DOCUMENT RESUME

ED 105 198 CE 003 486

AUTHOR	Stacey, William D.; And Others
TITLE	CODAP: Control Card Specifications for the Univac 1108.
INSTITUTION	Air Force Human Resources Lab., Lackland AFB, Tex. Computational Sciences Div.
REPORT NO	AFHRL-TR-74-84
PUB DATE	Oct 74
NOTE	206p.; For related volumes, see CE 003 485-487
EDRS PRICE	MF-$0.76 HC-$10.78 PLUS POSTAGE
DESCRIPTORS	Career Ladders; Comprehensive Programs; *Computer Programs; Information Processing; *Information Retrieval; Information Services; Information Sources; *Information Systems; Job Analysis; Occupational Clusters; *Occupational Information; Occupational Surveys; Task Analysis; *Technical Reports
IDENTIFIERS	Air Force; CODAP; Comprehensive Occupational Data Analysis Programs; UNIVAC 1108

ABSTRACT

The document is one of three in a series of technical reports covering the control card and programing aspects of the Comprehensive Occupational Data Analysis Programs (CODAP), a highly interactive and efficient system of computer routines for analyzing, organizing, and reporting occupational information. The document contains control card write-ups for the 34 main programs in the Univac 1108 version of CODAP. The write-ups are preceded by a short summary of the program functions. The summaries, and the write-ups which follow, are arranged alphabetically by program name. Each write-up includes detailed control card specifications necessary to use the CODAP system. Also contained in each write-up are general input-output specifications, general program logic, and diagnostic aids. (Author/AJ)

FEB 05 1975

AFHRL-TR-74-84

AIR FORCE HUMAN RESOURCES LABORATORY

ED105198

CODAP:
CONTROL CARD SPECIFICATIONS FOR THE UNIVAC 1108

By

William D. Stacey, MSgt, USAF
Johnny J. Weissmuller, Sgt, USAF
Bruce B. Barton, A1C, USAF
C. R. Rogers

U.S. DEPARTMENT OF HEALTH,
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

COMPUTATIONAL SCIENCES DIVISION
Lackland Air Force Base, Texas 78236

October 1974
Interim Report for Period January 1973 – September 1974

Approved for public release; distribution unlimited.

CE003486

AIR FORCE SYSTEMS COMMAND
BROOKS AIR FORCE BASE, TEXAS 78235

NOTICE

When US Government drawings, specifications, or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This interim report was submitted by Computational Sciences Division, Air Force Human Resources Laboratory, Lackland Air Force Base, Texas 78236, under project 7734, with Hq Air Force Human Resources Laboratory (AFSC), Brooks Air Force Base, Texas 78235. Dr. Raymond E. Christal, Chief of Occupational Research Division (OR), was project monitor, and Mr. William Phalen, OR, was the work unit scientist.

This report has been reviewed and cleared for open publication and/or public release by the appropriate Office of Information (OI) in accordance with AFR 190-17 and DoDD 5230.9. There is no objection to unlimited distribution of this report to the public at large, or by DDC to the National Technical Information Service (NTIS).

This technical report has been reviewed and is approved.

ROBERT A. BOTTENBERG, Chief
Computational Sciences Division

Approved for publication.

HAROLD E. FISCHER, Colonel, USAF
Commander

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFHRL-TR-74-84	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) CODAP: Control Card Specifications for the Univac 1108	5. TYPE OF REPORT & PERIOD COVERED Interim Jan 73-Sep 74
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) William D. Stacey Johnny J. Weissmuller Bruce B. Barton C. R. Rogers	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Computational Sciences Division Air Force Human Resources Laboratory Lackland Air Force Base, Texas 78236	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62703F 77340116
11. CONTROLLING OFFICE NAME AND ADDRESS HQ, Air Force Human Resources Laboratory (AFSC) Brooks Air Force Base, Texas 78235	12. REPORT DATE October 1974
	13. NUMBER OF PAGES 204
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Comprehensive Occupational Data Analysis Program
CODAP
Computer Programs
occupational survey
task analysis
hierarchical clustering
job descriptions
job types
FORTRAN
work analysis

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The Comprehensive Occupational Data Analysis Programs (CODAP) package is a highly interactive and efficient system of computer routines for analyzing, organizing, and reporting occupational information. Since its inception in 1960, CODAP has grown in tandem with advances in job analysis methodology and is now capable of answering most of the wide variety of management questions which confront CODAP users. This documentation of the UNIVAC 1108 CODAP system is being published in a series of 3 technical reports covering the

(SEE REVERSE SIDE)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Block 20. ABSTRACT (Con't)

control card and programming aspects of the system. A fourth report is in preparation by the Occupational Research Division of AFHRL which covers the research and operational applications of the CODAP system. This document contains control card writeups for the 34 main programs in the UNIVAC 1108 version of CODAP. These writeups are preceded by a short summary of the program functions. The summaries and the writeups which follow are arranged alphabetically by program name. Each writeup includes detailed control card specifications necessary to use the CODAP system. Also contained in each writeup are general input-output specifications, general program logic, and diagnostic aids.

PREFACE

The authors wish to express their gratitude to all those who have contributed to the preparation of this series of 3 technical reports:

AFHRL-TR-74-83 CODAP: Source Program Listings
AFHRL-TR-74-84 CODAP: Control Card Specifications
AFHRL-TR-74-85 CODAP: Programmer Notes

In particular, the following people have played a vital role in the development of these documents: Dr. Raymond E. Christal, Chief of the Occupational Research Division of AFHRL, Mr. William Phalen, also of that Division, and of course the secretaries, Betty L. Brooks and Laurel J. Betz, who not only typed but also helped in proofreading.

FOREWARD

The Comprehensive Occupational Data Analysis Programs (CODAP) System has been under continuous development for the past fifteen years. In its current form, it is a joint product of the Computational Sciences Division and the Occupational Research Division of AFHRL. In general, the Occupational Research Division has developed input-output specifications for programs and program modifications, while the Computational Sciences Division has provided programming services. However, even this distinction has not always been a clear-cut one, since suggestions for program improvements have come from individuals in both Divisions.

Since its conception, development of the CODAP system has been under the general direction of Dr. Raymond E. Christal, Chief of the Occupational Research Division. Dr. Christal also personally designed many of the programs and program modifications.

The earliest CODAP programs were written by Mr. Daniel Rigney and Mr. Wayne E. Fisher. This version was later expanded and reprogrammed (under contract) for execution on an IBM 7040 by Mr. Irwin R. Oats and Mr. James R. Hills of the Computer Sciences Corporation and later of Oats-Hills Incorporated, Houston, Texas. (Mr. Oats and Mr. Hills have translated CODAP programs for execution on the IBM 360 and 370 series computers, under contract with the U.S. Marine Corps and the Department of Defense.)

Since 1970 all CODAP innovations and modifications have been programmed in-house by the Computational Sciences Division. During this period, many new programs were added to the system, and nearly all old programs were extensively modified. During the last two years the CODAP system has been completely rewritten for execution on the Univac 1108. This reprogramming was first undertaken by MSgt William D. Stacey, with the assistance of Sgt Johnny J. Weissmuller. After MSgt Stacey's departure from AFHRL, the translation and reprogramming were completed and checked out by Sgt Weissmuller, assisted by A1C Bruce B. Barton. It is this version of the CODAP system which is documented in this technical report.

Many individuals have contributed to the CODAP system, and it would be difficult to specifically enumerate each contribution. However, the authors would be remiss not to mention those individuals who have participated in the mainstream effort. It is possible, with the passing of time and the failing of memories, that the following history contains serious omissions and inaccuracies, but it is correct in terms of currently available information.

The hierarchical clustering programs, OVRLAP, GROUP and KPATH, were initially designed by Joe H. Ward, Jr. and R. E. Christal, with original programming by Daniel Rigney and Wayne E. Fisher. The remainder of the first CODAP package was designed by R. E. Christal, with numerous suggestions from I. R. Oats, J. R. Hills and others. This pioneering effort included JOBGRP, JOBSPC, ASFACT, PRDICT, GRMBRS, PRTVAR, VARSUM, GRPSUM, GRPDIF, MXTPRT, TSETUP, INPSTD, VARGEN, EXTRCT, and JOBINV. The programming for these was accomplished by I. R. Oats, J. R. Hills and D. W. Hartmann.

The table below lists the programs developed after the initial package:

Program	Initial Design	Initial Programming
SETCHK	W. L. Wells	Edward L. Adams
PROGEN	Edward L. Adams	Edward L. Adams
TSKNDX	R. E. Christal	William D. Stacey
AVALUE	R. E. Christal	William D. Stacey
DIST2X	C. R. Rogers	Johnny J. Weissmuller
CORREG	Joe H. Ward Jr. Robert A. Bottenberg	Janice Buchhorn Kathleen Hall William D. Stacey
AUTOJT	William Phalen	Paul Aron
PRIJOB	William Phalen R. E. Christal	Computer Sciences Corporation
DUVARS	William Phalen	William D. Stacey
REXALL	William Phalen Johnny J. Weissmuller R. E. Christal	Johnny J. Weissmuller
AVGPCT	William Phalen	Johnny J. Weissmuller
JDINDX	Johnny J. Weissmuller	Johnny J. Weissmuller
JOBIND	R. E. Christal	Edward L. Adams
TSKGRP	Harry Lawrence Johnny J. Weissmuller	Johnny J. Weissmuller
DIAGRM	William Phalen R. E. Christal Philip Aitken-Cade	Computer Sciences Corporation

Among other individuals who have played a meaningful part in the development, improvement and maintenance of the CODAP system, are the following:

Dr. Robert A. Bottenberg Mr. Jim Souter Mr. C. R. Rogers	Monitoring and Direction of Programming
Mr. J. Myer	Systems and Programming
Joe Morsh Wayne Archer	Numerous Suggestions for Program Changes
Harry Kudrick W. L. Wells Robert L. Vance Bill Feltner, Jr. Wesley C. Hill Steve D. Poole Terry D. Lewis	Program Execution and Suggestions for Improvements

INTRODUCTION:

PHASES OF A CODAP STUDY

I. DATA PREPARATION

A. Generally this step includes utility programs for sorts, merges, etc. May include optical scanning and reformatting routines.

B. Final run will be a SETCHK run to produce either the Raw-Data file for INPSTD, or a Rater file for either REXALL or TSKNDX.

II. CODAP FILE CREATION

A. INPSTD creates the History Data file from the Raw-Data file and the control cards. Cases are sequenced the same as on the Raw-Data file and some variables (C003-6, C009) are left zero, to be filled in by KPATH.

B. TSETUP initializes the Job Description and/or Report files. Each of these files consists of 3 reels of tape. Each reel is a copy of the file, and each copy is called a file cycle. Cycles are denoted (0), (-1), and (-2). Cycle (0) is the most current copy and cycles (-1) and (-2) are the first and second backup, respectively. Only JOBGRP and JOBSPC are able to add records to the Job Description file, while many programs may add reports to the Report file. (See SECTION VI B.)

III. HIERARCHICAL CLUSTERING (Optional)

A. OVRLAP computes the similarity between all pairs of cases and creates the OVRLAP Matrix file.

B. GROUP uses the OVRLAP Matrix file to hierarchically cluster cases into larger and larger groups. Each merger is called a stage and each stage uniquely defines a group of individuals which was created at that stage. The output file from GROUP is called the Stage Data file.

C. KPATH uses the information on the Stage Data file to reorder the cases on the History Data file. The History Data file is reordered, additional information is inserted into each case record (C003-6, C009) and the KPATH file is written out. The KPATH file may be used by any program which could use the History Data file, but the converse is not necessarily true. (See JOBGRP in SECTION V B.)

D. GRMBRS reads the KPATH file, produces a report on the group mergers, and creates the GRMBRS file to be used by DIAGRM.

E. DIAGRM reads the GRMBRS file and prints a pictorial representation of the hierarchical grouping process.

IV. FILE MODIFICATION

A. These programs may be executed at any time after INPSTD or KPATH (depending on the file used). When and if they are used is purely at the discretion of the task scientist or project director.

B. DUVARS creates a new History/KPATH file with additional background (VXXX) variables based upon duty performance.

C. VARGEN creates a new History/KPATH file with additional computed (CXXX) variables based upon options using a combination of a weight deck and the individual's task values.

D. PROGEN is a FORTRAN program generator. It generates a FORTRAN source deck which will handle the overhead regions of a History/KPATH file, allowing the expert CODAP user to create, modify, or use either background or computed variables. NOTE: PROGEN merely creates a source deck. It is left to the user to produce an executable program from that deck.

V. GROUP IDENTIFICATION

A. JOBSPC will identify all individuals with specified characteristics based upon the values of either background or computed variables. Either a History or KPATH file may be used, but if hierarchical clustering is to be done at all, it is recommended that only the KPATH file be used. Optionally, the report may be copied to the Report file and this group recorded by adding an entry on the Job Description file. This allows the group to be identified later by merely inserting the request card which will be punched.

B. JOBGRP will identify all individuals who were in the group that was formed at a specified stage of the grouping process. Note that only the KPATH file may be used as input as the cases are not properly sequenced on the History file, and requisite information would not be available. Also, the function of JOBGRP may be absorbed by JOBSPC inasmuch as specifying a range on C009 (KPATH SEQUENCE NUMBER) can identify a merging group and the range may be found on either the GRMBRS or DIAGRM outputs. JOBGRP, like JOBSPC, may optionally add reports to the Report file and make entries on the Job Description file.

VI. REPORT PROGRAMS

NOTE: A preceding asterisk (*) denotes the program requires the Job Description file.

A. INTERMEDIATE REPORTS: Cannot be added to Report file.

1. CORREG computes correlations and regressions on up to 100 computed or background variables.

* 2. AUTOJT produces measures of group differences on up to 850 pairs of groups.

3. DIST2X prints a two-way distribution on specified pairs of computed or background variables.

* 4. JDINDX lists an index of all entries on a Job Description file.

5. VARMEN computes means and standard deviations on up to 100 computed or background variables. (Reduced version of CORREG.)

B. INTERMEDIATE OR FINAL REPORTS: May be added to Report file.

* 1. ASFACT distributes task ratings on secondary factors.

2. AVALUE computes the mean of a given variable for each task based on the cases which performed that task.

3. AVGPCT is the same as AVALUE except that the percent of a group performing is used, rather than the actual number of cases. This will tend to reduce sampling bias.

* 4. GRPDIF prints a task-by-task difference between groups.

* 5. GRPSUM prints task summaries of several groups per page.

* 6. JOBIND prints job descriptions for each case in a given group.

7. JOBINV prints all the task and duty statements in the inventory.

* 8. MTXPRT prints an overlap matrix for specified groups.

9. PRDICT prints the variable dictionary which identifies all the computed (CXXX) and background (VXXX) variables.

* 10. PRIJOB prints the primary job identifiers such as the most performed or most time consuming tasks.

* 11. PRTVAR sorts and prints variables for all individuals in specified groups.

12. REXALL computes the INTER-RATER Reliability coefficient from the Rater Data file, and optionally punches a weight deck.

13. TSKGRP reports the task index values from a Rater Data file or a weight deck for a specified set of tasks and a specified group of cases.

14. TSKNDX reports the task index values from a Rater Data file or a weight deck for a specified group of cases.

* 15. VARSUM will distribute the values of specified variables for specified groups.

DESCRIPTION OF CODAP PROGRAMS

ASFACT: REPORTING SECONDARY FACTOR DATA

If the tasks in a job inventory are rated on a second scale such as "amount of training required," the data is handled through the ASFACT program. The ASFACT program reports the following information on each task for any group of individuals whose job description has been computed by JOBGRP or JOBSPC: (1) frequency distribution of members responding on the secondary factor (0 to 9 scale); (2) total number of respondents; (3) number of respondents with values outside the specified range; (4) arithmetic mean and standard deviation of acceptable responses.

AUTOJT: AUTOMATED JOB TYPE SELECTION PROGRAM

This program calculates, evaluates, and reports between-group differences for specified pairs of groups whose job descriptions were computed by JOBGRP or JOBSPC. Six comparison options are used: (1) differences in average percent time spent on each task; (2) differences in average percent time spent on each duty; (3) differences in percentage of members performing each task; (4) differences in number of tasks accounting for a major portion of average percent time spent; (5) differences in number of duties accounting for a major portion of average percent time spent; (6) differences in average number of tasks performed. As many as 850 pairs of groups can be compared in one run of AUTOJT.

AVALUE: CALCULATING AVERAGE VALUE

For each task in a job inventory, this program calculates and prints the mean and standard deviation of a selected background or computed variable using all valid responses of individuals in a specified group who perform the task. Task titles and the number of valid respondents on each task are also reported. Average values are optionally sequenced from low-to-high average value, high-to-low average value, or in task number order. AVALUE can be calculated for any group of individuals who can be identified on some range of a background or computed variable or by the intersection of variables.

AVGPCT: AVERAGE BY PERCENT PERFORMING

This program is a version of the AVALUE program - AVGPCT computes an average level which is based on the percent of members in each level actually performing each task and AVALUE computes simply the mean value associated with each task.

CORREG: CORRELATION AND REGRESSION PROGRAM

This program package extracts up to 100 computed and background variables from a CODAP KPATH or HISTORY file and computes correlation matrices and regression problems. The correlation program computes and prints the correlation matrix, number of valid and invalid cases in the sample, and means and standard deviations of variables. These same computations are also put on file for future reference, and they also remain in core if they are to be fed into the regression routine immediately. In this routine, a series of regression problems is computed iteratively. Beginning with the best combination of three predictors at iteration 1, the best remaining combination of three predictors is added to the model at each subsequent iteration. The "best remaining combination" is that which adds the most to the value of R^2 when used in conjunction with all combinations of predictors selected at previous iterations. Predictors may be used more than once during the iterative process. Iterations are continued until the increase in the value of R^2 over the previous iteration is less than some amount specified by the requester. Variable ID's, R^2 values, and the error sum of squares are reported for each iteration. The standard and raw score weights for each variable as they exist upon completion of the final iteraiton are reported, as well as the regression constant.

DIAGRM: GRAPHICAL PRESENTATION OF HIERARCHICAL GROUPING ACTIONS
This program used data from the GRMBRS program to generate a treelike diagram that visually displays the order in which groups merged during the hierarchical grouping process. Each group is represented by a rectangular block of data containing the same information found in GRMBRS. Rows and columns of asterisks show the branches leading from a group to its subgroups. Control cards can be used to limit the number and type of groups displayed by DIAGRM.

DIST2X: COMPUTING A TWO-WAY DISTRIBUTION
This program distributes a group of individuals on two variables (a row variable and a column variable). The group distributed may be the total sample or the cases coded "1" or "0" on a categorical membership variable. The row and column variables may be alpha or numeric, and intervals of unequal width may be defined for any variable. The data presented may be frequencies within cells or frequencies and percentages within cells. The percentages are computed based on individual row frequency totals, or on individual column frequency totals, or on the total N for all rows and columns. Any or all three sets of percentages may be displayed in the two-way table. A "total" category for rows and a "total" category for columns is automatically included for percentages and/or frequencies. The total number of cases counted and the total number of valid cases are given for each rwo and column and for total rows and total columns. Valid cases include all distributed numeric data. Only valid cases are used in the computation of means and standard deviations. Optionally,

an "other" category may be included for rows and columns. Inclusion of the "other" category causes all numeric data to be counted as valid. Another option provides for the computation of means and standard deviations for individual rows and total rows and/or individual columns and total columns. Only cases defined as valid are used in computing means and standard deviations. A summary of selected information appears at the end of the report. Any number of reports may be generated in one program execution.

DUVARS: COMPUTING A DUTY VARIABLE FOR EACH CASE

This program uses task data to compute duty values for each case and displays the duty values in the form of a PRTVAR. Three duty variable options are available: (1) percent time spent in each duty by each case; (2) number of tasks performed in each duty by each case; (3) percent of tasks performed within each duty by each case as a function of his total number of tasks performed. This program is used principally as an aid in selecting meaningfully different job-type groups.

EXTRCT: REORDERING AND EXTRACTING REPORTS FROM THE REPORT FILE

All reports generated by CODAP can optionally be placed on the REPORT file for future recall. The EXTRCT program extracts reports selected by control cards from the REPORT file, prints them out in the same order as the input requests with continuous page numbering, and furnishes a table of contents with page number references. An option is available to provide an 'INDEX' of the REPORT file.

GRMBRS: REPORTING GROUP MEMBERSHIP

This program produces an information report that identifies the two groups combining at each stage of the hierarchical grouping process. The information includes the stage number, the number of members in the combined group, the number of members in each of the combining groups, range of KPATH sequence numbers for the combining groups, the average percentage of overlap between the members of the combining groups, and the average percentage of overlap within the combined group.

GROUP: CLUSTERING INDIVIDUALS AND GROUPS OF INDIVIDUALS

This program uses the similarity matrix computed in the overlap (OVRLAP) program to form clusters of cases. The grouping technique, called "collapsing the matrix" or "hierarchical grouping," involves repeated searching for those individuals or partially formed clusters which have the highest (or lowest) remaining similarity, depending on whether a "maximizing" or "minimizing" process was requested. The "maximizing" option is always used for job survey data. Each new clustering or "collapse" is called a "stage" and the vectors of similarity values for the clusters being merged are combined according to a specified mathematical algorithm to form an integrated cluster. Several formulas for combining groups are available. The collapsing process continues until a single group has formed which contains all cases in a study.

GRPDIF: DIFFERENCE COMPARISON BETWEEN JOB DESCRIPTIONS

This program calculates and reports the difference between two job descriptions in terms of percentage of members performing each task and/or average percent time spent. Difference values are presented in ascending or descending order on either value (from largest negative to largest positive difference or vice versa) or in task number order.

GRPSUM: SUMMARIZING JOB DESCRIPTIONS

This program calculates and prints a report of either the percent of members performing each task in the job inventory or the average percent time spent on each task by all members for any number of groups whose composite job descriptions were computed by JOBGRP or JOBSPC. The summarized data is printed in task number order and the group descriptions are ordered according to the sequence of the input request cards.

INPSTD: RAW DATA EDITING AND INPUT

This program reads task titles, task responses, and background data from tape and card input. It converts the raw task responses to percentages, constructs data vectors for each case, reorganizes the data to a standard data format, and writes the formatted data on the HISTORY file for use in subsequent programs. INPSTD will accept a maximum of: 1700 background variables, 1700 task variables, 26 duty variables, and 20,000 cases.

JDINDX: PRINTING AN INDEX OF THE JOB DESCRIPTION FILE

This program prints information for every report found on the JOB DESCRIPTION file. Information printed is: study ID, NCASE, NDUTY, Filename, NTASK, Report ID, Source file, number of members, and Report number.

JOBGRP: CALCULATING COMPOSITE JOB DESCRIPTIONS

This program calculates and prints composite job descriptions for groups formed during the hierarchical grouping process. Both duty and task job descriptions may be reported in high to low sequence of either "average percent time spent by all members" or "percent of members performing." (A duty is a functional area comprising a number of tasks and, hence, is a summary report.) A job description provides the following information: duty/task number, duty/task title, percent of members performing each duty/task, average percent time spent by members performing, average percent time spent by all members, and cumulative average percent time spent by all members. JOBGRP is limited to 7000 cases since that is the maximum allowed by OVRLAP/GROUP.

JOBIND: CALCULATING INDIVIDUAL JOB DESCRIPTIONS

The JOBIND program calculates and prints job descriptions for individual cases. Output will be task statements sequenced from high to low percent time spent, together with their percent time spent values and cumulative percentages. Selected background information may also be printed at the top of each description. JOBIND has six options for selecting cases: (1) by last two digits of social security number; (2) by every Nth case beginning with a specified KPATH sequence number; (3) by meeting specified requirements on background variables; (4) by membership in a specified job type group; (5) by case control number; (6) randomly, using a random number generator. The selected descriptions may be sorted in KPATH order, in random order, by background variable, or in case control number order. The six options for selecting cases may also be used to create new categorical membership variables which are inserted in the variable dictionary as computed variables.

JOBINV: PRINTING OF DUTY AND TASK TITLES
This program prints a listing of the duty and task titles included in a job inventory. The titles are listed in task number sequence within each duty, and the format calls for two columns per page.

JOBSPC: CALCULATING COMPOSITE JOB DESCRIPTIONS
This program calculates and prints composite job descriptions for "special" groups whose membership is defined in terms of some combination of background or computed variables. Both duty and task job descriptions may be reported in high to low sequence of either "average percent time spent by all members" or "percent of members performing." (A duty is a functional area comprising a number of tasks and, hence, is a summary report.) A job description provides the following information: duty/task number, duty/task title, percent of members performing each duty/task, average percent time spent by members performing, average percent time spent by all members, and cumulative average percent time spent by all members.

KPATH: ORDERING A HISTORY FILE
After OVRLAP and GROUP have been completed, the KPATH program assigns sequence numbers to individual cases in such a way that each pair of individuals or groups which are merged during the grouping process will have a contiguous block of KPATH sequence numbers. This program creates the KPATH file.

MTXPRT: PRINTING AN OVERLAPPED MATRIX
This program calls for the subroutine OVRLAP to overlap all possible pairings of a set of composite job descriptions and then uses the program MTXPRT to print the between-group overlap values in matrix form. Overlap may be computed in terms of average percent time spent on tasks or in terms of number of tasks performed in common. The maximum number of groups that can be input to MTXPRT is 100.

OVRLAP: RELATING RESPONSES TO EACH OTHER
This program generates an overlap or similarity matrix of all possible paired comparisons between individual cases. Similarity is expressed as a percentage of common tasks performed or as the total overlapping percentage of time spent on tasks. Overlap in terms of percent time spent is the preferred option in most studies. OVRLAP is restricted to 7000 cases.

PRDICT: PRINTING DICTIONARY OF VARIABLE TITLES
This program prints a report containing the identification codes and descriptive titles of all background and computed variables peculiar to a particular study. The identification codes are used in calling for data to be reported or to be acted upon by various programs in the CODAP system.

PRIJOB: CALCULATING PRIMARY JOB IDENTIFIERS
This program calculates and prints a report of those tasks which are determined to be "primary identifiers" of job types. Primary identifiers may be defined as the top x-number of tasks in a group job description in terms of percent of members performing or average percent time spent by all members. Primary identifiers may also be defined as those tasks performed by a specified minimum percentage of the group members or those which exceed a specified average percent time spent value. This program allows a number of groups to be aligned in a single report for comparative purposes.

PROGEN: PROGRAM GENERATION PROGRAM
This program permits the CODAP expert to add, extract, and manipulate data in the CODAP system in ways not encompassed by the standard CODAP programs and do so with a minimum amount of additional programming and without requiring thorough indoctrination in the CODAP system. PROGEN uses a combination of FORTRAN statements and shorthand operation codes to generate a special purpose program to read the KPATH or HISTORY files, either of which contains the entire data for each case, and perform operations upon it. New variables can thereby be created and added to the CODAP variable dictionary. This program also has the facility to reconvert percent time task values for each case to the original raw response form, perform operations on the raw responses, and convert them back to percentages.

PRTVAR: PRINTING VARIABLES
PRTVAR enables the user to select variables and print-out formats to produce a report of the case data values for the selected background and computed variables. The data may be sorted and groups may be selected using request cards.

REXALL: COMPUTING INTERRATER RELIABILITY
This program computes and reports for a group of raters the average interrater reliability coefficient of a single rater and the stepped up reliability coefficient for the total group of raters. The program is most often used in conjunction with sets of task difficulty ratings made by a large number of supervisory personnel. The program computes and reports the distribution, mean, and standard deviation of the mean task ratings and the standard deviations of the ratings for each task.

SETCHK:	EDITING RAW DATA This program makes certain checks to raw data to insure valid input to the CODAP system. It range checks response data, insures that case data is complete and in the proper sequence, and makes tests to determine if the data meets certain other requirements of the system. Invalid data is deleted from the file.
TSETUP:	INITIALIZE THE JOB DESCRIPTION AND REPORT FILES This program writes information at the head of these files to prepare them for the accumulation of reports and statistical information which will occur throughout the system.
TSKGRP:	TASK INDEX RESTRICTED TO SPECIFIED TASKS This program is nearly identical to TSKNDX except that the reports are limited to a specified group of tasks. In addition, a cumulative summary of all reports generated is given at the end of each run. TSKGRP is primarily used to compute the difficulty of and percent time spent on a given group of tasks by a specified group of individuals.
TSKNDX:	CALCULATING AVERAGE TASK RATINGS This program has the same options as AVALUE, except that it is used with task ratings rather than with background or computed variables. TSKNDX can also provide the following additional task information not available in AVALUE: (1) percent of members performing; (2) average percent time spent by members performing; (3) average percent time spent by all group members; (4) cumulative sum of average percent time spent by all group members. TSKNDX is primarily used to compute average task difficulty ratings.
VARGEN:	COMPUTING NEW VARIABLES This program calculates new computed variables by applying input data to the task values of each case. The task values may be time spent percentages or "do - don't do" values (1,0). The input data consists of a vector of weights, a scaling (standardizing) factor and a specified calculation formula (four options). A newly created variable is given a variable identification number and is added into the computed variable portion of the case data records on the HISTORY or KPATH files.
VARMEN:	VARIABLE MEANS AND STANDARD DEVIATIONS This program selects cases with valid values for all specified variables (up to 100) then reports means and standard deviations on these cases.
VARSUM:	SUMMARIZING BACKGROUND AND COMPUTED VARIABLES This program computes and reports frequency distributions within specialized intervals, makes total frequency counts, and calculates means and standard deviations on selected background and computed variables for any group of individuals whose job description has been generated by JOBGRP or JOBSPC.

PROGRAM IDENTIFICATION

Name	ASFACT (Analysis of Secondary Factors)
Language	FORTRAN V
Date Written	1966/April 1974
Programmer	CSC/Weissmuller

FUNCTION

Produces a report of secondary factor data for job description members selected by programs JOBGRP or JOBSPC. The report will include: number of members in the job description group; number of members in the factor subgroup; frequency distribution of members responding to secondary factors (0-9 scale) by task; total responding; number outside the specified range; arithmetic mean; and standard deviation of members within the range.

INPUT/OUTPUT

Input	a. HISTORY/KPATH file
	b. JOB DESCRIPTION file
	c. REQUEST cards
	d. SETCHKed Factor Data file
Output	a. Printed report
	b. Printed report optionally added to the REPORT file.

CONTROL

Section I FILENAMES CARD

Columns	
1- 6	ASFACT
8-18	Filename of the HISTORY/KPATH file
20-30	Filename of the SETCHKed Factor Data file
32-42	Filename of the JOB DESCRIPTION file
44-54	Filename of the REPORT file

Section II IDENTIFICATION CARD

Columns	
1- 6	ASFACT
8-15	Study ID

Section III CONTROL CARD

Columns	
1- 6	REPORT ID
8-55	REPORT TITLE
57	Number of PAGE HEADING cards to follow in Section IV. 0-6 cards are allowed.
59-60	Starting card number (cc 74-75) of factor data which contains task rating data. There must be NTASK ratings present. They must be in cc 5-73 of each card for as many cards as required. Background information cards may precede or follow rating data but will be ignored.
62	Minimum datum value to be tallied in the frequency distribution (numeric 0-9). Assumes 1 if blank.
64	Maximum datum value to be tallied in the frequency distribution (numeric 0-9). Assumes 7 if blank.
66	1 = Do not add this report to the REPORT file. 0 = Add this report to the REPORT file.
68	1 = Do not print the frequency distribution. 0 = Print the frequency distribution of data defined by high and low limits.
70	1 = Do not print the number of 'other'. 0 = Print the number of members outside data value limits given.

72 1 = Do not print the total cases counted, MEANS, S.D.'s.

0 = Print the total cases counted, MEANS, S.D.'s.

Section IV PAGE HEADING CARDS

Columns

1-80 Any page heading information desired. 0-6 cards are allowed.

Section V REQUEST CARDS

Columns

1-80 These cards were produced by JOBGRP or JOBSPC. A maximum of 12 cards is allowed.

Section VI END CARD

Columns

1- 3 END

NOTE: Multiple runs may be made using Sections III-VI.

GENERAL DESCRIPTION

The HISTORY/KPATH file is read and the titles and certain control information is saved. The JOB DESCRIPTION file is read and a NMASK array is constructed to indicate which cases are needed on the secondary factor file. The factor data information is read and accumulated for the requested job descriptions. The report is printed and added to the REPORT file if requested.

MESSAGES

1. 'MORE THAN SIX HEADING CARDS SPECIFIED, RUN ABORTED.' Self explanatory.
2. 'REPORT NUMBER XXX WAS NOT CREATED WITH THE KPATH/HISTORY FILE USED WITH THIS RUN. RUN ABORTED.' If the KPATH file was used to create this report, it must be used for this run. If the HISTORY file was used to create this report, it must be used.

3. CANNOT LOCATE REPORT NUMBER XXX. RUN ABORTED.' The JOB DESCRIPTION file does not contain the requested report.

4. 'STUDY ID NO MATCH. TAPE IS XXXXXXXX.' Self explanatory.

PROGRAM IDENTIFICATION

Name	AUTOJT (Automated Job Typing)
Language	FORTRAN V
Date Written	1971/May 1973
Programmer	Aron/Weissmuller

FUNCTION

This program calculates, evaluates, and reports between-group differences for specified pairs of groups whose job descriptions were computed by JOBGRP or JOBSPC. Six comparison options are used: (1) differences in average percent time spent on each task; (2) differences in average percent time spent on each duty; (3) differences in percentage of members performing each task; (4) differences in number of tasks accounting for a major portion of average percent time spent; (5) differences in number of duties accounting for a major portion of average percent time spent; (6) differences in average number of tasks performed. As many as 850 pairs of groups can be compared in one run of AUTOJT.

INPUT/OUTPUT

Input	a. HISTORY/KPATH file
	b. JOB DESCRIPTION file
	c. REQUEST cards
Output	a. Printed report

CONTROL

Section I FILENAMES CARD

Columns	
1- 6	AUTOJT
8-18	Filename of the HISTORY/KPATH file
20-30	Filename of the JOB DESCRIPTION file

Section II CONTROL CARD 1

Columns	
1- 6	AUTOJT
15	1 for Option 1
16	2 for Option 2 (Cumulative % Time Spent - Tasks & Duties)
17	3 for Option 3 (Average Number of Tasks Performed)
18	1 for Option 1 - Variant 1 (Percent Time Spent on Tasks)
19	2 for Option 1 - Variant 2 (Percent Time Spent on Duties)
20	3 for Option 1 - Variant 3 (Percent Members Performing Tasks)
21-30	C Constant for Option 1 Variant 1 (F10.0) - assume 39.0625
31	1 for |D|, 0 or b̸ for D**2 Option 1 - Variant 1
32	D, G, or W for denominator type - assume G Option 1 - Variant 1
33-42	D, G, or W Value (F10.0) - assume 1.00 Option 1 - Variant 1
43-52	K value - assume 1.00 Option 1 - Variant 1
53-55	CUTV for Option 1 - Variant 1 (I3) assume 5, i.e., the program will stop after detecting decreasing E Values for CUTV iterations.
56-65	XCUTV value for Option 1 - Variant 1 (F10.0), i.e., stop iterations when the D value for this iteration is less than this number. Assume 0.00.

66-75	C Constant for Option 1 - Variant 2 (F10.0) assume 156.25
76-80	PRINT for complete heading information in front of each Option in the iterative print out section (assumed constants, denominators, etc.) -- this generates so many pages of output that the assumed option (leave this field blank) skips printing these assumed and calculated values after the first pair.
Section III	CONTROL CARD 2 (May be blank)
Columns	
1	1 for |D|, 0 or ƀ for D**2 for Option 1 - Variant 2
2	D, G, or W for Option 1 - Variant 2 (F10.0) assume G
3-12	D, G, or W value for Option 1 - Variant 2 assume 1.00
13-22	K value for Option 1 - Variant 2 - (F10.0) assume 1.00
23-25	CUTV for Option 1 - Variant 2 (I3) assume 5
26-35	XCUTV for Option 1 - Variant 2 (F10.0) assume 0.00
36-45	C Constant for Option 1 - Variant 3 (F10.0) assume 10000.0
46	1 for |D|, 0 or ƀ for D**2 Option 1 - Variant 3
47	D, G, or W for Option 1 - Variant 3
48-57	D, G, or W value for Option 1 - Variant 3 (F10.0) assume 1.00
58-67	K value for Option 1 - Variant 3 (F10.0) assume 1.00
68-70	CUTV for Option 1 - Variant 3 (I3) assume 5
71-80	XCUTV for Option 1 - Variant 3 (F10.0) assume 0.00
Section IV	CONTROL CARD 3 (May be blank)
Columns	
1-10	K Constant for Option 2 (F10.0). Assume 1.00 for percent time spent on tasks, 1.00 for percent time spent on duties.

11-20	C Constant for Option 2 (F10.0) assume 12.50 percent time spent on tasks, 18.75 for percent time spent on duties.
21-30	G Constant for Option 2 (F10.0) assume 1.00 for tasks and duties.
31-33	CUTV for Option 2 (I3) stop iterations after the program detects decreasing E Values CUTV times. Assume 5
34-43	K Constant for Option 3 (F10.0) assume 1.50 Average Number of Tasks Performed
44-53	G Constant for Option 3 (F10.0) assume 1.00
Section V	REQUEST CARDS (In pairs)
Columns	
1-80	These are the system produced REQUEST cards. 850 pairs are allowed.

GENERAL DESCRIPTION

The control cards are read in and scanned for errors. A table of the JSEQUENCE numbers of the requested reports is constructed and the report sequence of the last-requested job description is determined. The JOB DESCRIPTION file is read, and if a description was requested, it is placed on a disk file. When the last-requested job description has been copied onto the disk file the program begins its iterative loop. Each Option is checked for in turn with respect to each of the pairs of requested job descriptions which are randomly retrieved from the disk. For each specified calculation option constants are calculated -- if the control card field was left blank the assumed value is supplied by the program. Iterations can be terminated in one of three ways for the calculation options. The D Value may go below a specified cutoff value (XCUTV). The iterations may be carried on N iterations (CUTV) with decreasing E VALUEs. Finally, the program may exhaust all of the non-zero task responses for any given calculation option and quash further iterations--if this does occur, the program prints a message informing the user that all of the non-zero data elements have been exhausted. After all of the pairs of job descriptions have been examined and the iterative information has been printed in full the program also prints an "E VALUE SUMMARY." The summary contains information on peak E Values for each of the specified options, the iteration on which this peak was reached, the pair number of this comparison, and the page on which it may be found. The summary is printed in three sort sequences: Sorted on the numeric portion of Group A's ID, then on the numeric portion of Group B's ID, and then in the original request-card input sequence. In each of the three summary sections E VALUES which are greater than 1.00 are flagged by three or four dollar signs ($$$ or $$$$).

MESSAGES

1. 'NO LEGAL COMPUTATIONAL OPTION(S) SPECIFIED - PROCESSING TERMINATED.' None of the Options were called for in columns 15 through 17 of control card number 1.
2. 'SPECIFICATION ERROR - NNNN JSEQUENCE NUMBERS NOT FOUND ON THIS INPUT TAPE - PROCESSING TERMINATED.' Requested a job description which cannot be located.

PROGRAM NOTES

OPTION 1 -

$$E = \frac{\sum_{i=1}^{k} A_i / \sqrt{b}}{c * d}$$

where $A_i = |D_i|$ or D_i**2, as specified. D_i is the difference value for: Variant 1 - Percent Time Spent on Tasks (assume D**2)

Variant 2 - Percent Time Spent on Duties (assume D**2)

Variant 3 - Percent Members Performing Tasks (assume D**2) b = the number of A_i values summed at any given iteration.

c = a constant: assumed values are 39.0625, 156.25, and 10000.0 for Variants 1 through 3.

d = a constant which may be one of three types:

TYPE D = specified value (assumptions are 1.0, 1.0, and 1.0)

TYPE W = specified constant used in the formula

$$d = \frac{W}{MIN\ (X,\ Y,\ W)}$$

where X and Y are the sizes of the two groups being compared.

TYPE G = a specified constant used in the formula

$$d = \sqrt{\frac{G * (X + Y)}{X * Y}} + K$$

this is the assumed constant type.

OPTION 2 -

$$E = \frac{\sum_{i=1}^{k} Ta_i - \sum_{i=1}^{k} Tb_i}{K + \sqrt{\frac{g * (X+Y)}{X * Y}}}$$

where Ta_i and Tb_i are the percent time spent values for groups a and b respectively (first on Tasks, and then on Duties).

g = specified Constant (assume 12.50 for tasks, 18.75 for duties)

K = specified Constant (assume 1.00)

OPTION 3 -

$$E = \frac{M_L\ /\ M_S}{c}$$

where M_L = the average number of tasks performed by the group performing the larger average number of tasks.

M_s = the average number of tasks performed by the group performing the smaller average number of tasks.

$$c = K + \sqrt{\frac{G * (X + Y)}{X * Y}}$$

where K = a specified value (elevation adjustment) assume 1.50

G = a specified value (slope adjustment) assume 1.00

X = number of members in GROUPA.

Y = number of members in GROUPB.

PROGRAM IDENTIFICATION

Name	AVALUE (Average Value)
Language	FORTRAN V
Date Written	1970/May 1973
Programmer	Stacey

FUNCTION

For each task in a job inventory, this program calculates and prints the mean and standard deviation of a selected background or computed variable using all valid responses of individuals in a specified group who perform the task. Task titles and the number of valid respondents on each task are also reported. Average values are optionally sequenced from low-to-high average value, high-to-low average value, or in task number order. AVALUE can be calculated for any group of individuals who can be identified on some range of a background or computed variable or by the intersection of variables.

INPUT/OUTPUT

Input	a.	HISTORY File or KPATH File.
	b.	Control Cards.
Output	a.	Printed report.
	b.	Full report on the REPORT file if requested.
	c.	Average value weight deck if requested.

CONTROL

Section I: FILENAMES CARD

Columns	
1-6	'AVALUE'

CONTROL (continued)

Section I:

Column	
8-18	Filename of the HISTORY or KPATH file.
20-30	Filename of the REPORT file.

Section II: IDENTIFICATION

Column	
1-6	'AVALUE'
8-15	Study ID on the KPATH or HISTORY file.

Section III: CONTROL

Columns	
4-7	A computed or background variable which is the 'average value' variable. Cannot use computed variables C001, C002, or C003. Cannot use a computed variable whose value could exceed 999. Background variables cannot exceed three characters in length and must be numeric.
9	Specify the field size of the 'average value' background variable. Maximum is three characters. Blank if column 4 is 'c' or if column 11 is '1.'
11	1 = The variable specified in columns 4-7 is AFSC and the skill level position only is wanted as the 'average value.' The AFSC variable must be a five character field such as '51170.'

CONTROL (continued)

Section III:

Columns	
	0 = The variable is not AFSC
13	Sort option for printed report. 1 = Ascending on average value 2 = Descending on average value 3 = Duty/task order
15	Specify the number of PAGE HEADING cards which follow. Must be between 1 and 6.
17	0 = Punch a weight deck. This deck contains 'average values' in duty/task order. There will be NTASK values punched 10 to a card in F7.2 Format. 1 = Do not punch a weight deck.
19	0 = Add this report to the REPORT file. 1 = Do not add this report to the REPORT file.
21-26	Special ID of this report. ID is written on the REPORT file and punched into the REQUEST card. It also appears on the page number line of all report pages.
28-32	Specify the low range for the 'average value' variable. If the 'average value' variable is less than this number, the case is bypassed. This is a F5.0 field. If the decimal point is not punched, the value must be right adjusted in its field. This field is required.

CONTROL (continued)

Section III:

Columns	
34-38	Specify the high range for the 'average value' variable. If the 'average value' variable is greater than this number, the case is bypassed. This is a F5.0 field. If the decimal point is not punched, the value must be right adjusted in its field. This field is required.

Section IV: TITLE CARD

Columns	
1-48	This information is printed on the page number line of each page of the report. This field is punched into the REQUEST card if the report is added to the REPORT tape. This card is required even if it is blank.

Section V: PAGE HEADING CARD(S)

Columns	
1-80	Any information the user desires. PAGE HEADING cards appear on the first page of the report. Must have between 1 and 6 PAGE HEADING cards.

Section VI: SAMPLE SELECTOR CARDS

Refer to 'SAMPLE SELECTION' in this manual for a complete description of sample selection procedures.

Note: Sections III, IV, V, and VI may be repeated for multi-report runs.

GENERAL DESCRIPTION

The program reads the communications region from the KPATH or HISTORY file and computes the number of reports that it can generate with one pass of the input file. Control cards are read until EOF is encountered or until IMAX 'sets' have been read. Data cases are read and tested to see if they fall within the range specified by selector variables for any of the 'sets.' The 'average value' variable field is range checked and summed into arrays for selected cases. When the input has been passed, means and standard deviations are computed, and the weight decks are punched if requested. Average values are sorted according to control card options and the reports are printed and also added to the REPORT file if requested. If the program has encountered EOF, it terminates. If not, it recycles.

MESSAGES

1. 'INCORRECT SORT SEQUENCE FOR XXXXXX. ASSUMED 1.' Read Section III, column 13.
2. 'S.I.D. NO MATCH. RUN TERMINATED.' Study I.D. from control card does not match Study I.D. on tape.
3. 'THE NUMBER OF PAGE HEADING CARDS SPECIFIED FOR XXXXXX IN ERROR. RUN TERMINATED.' Read Section III, column 15.
4. 'WEIGHT OPTION INCORRECTLY SPECIFIED FOR XXXXXX. ASSUMED 0.' Read Section III, column 17.
5. 'REPORT FILE OPTION INCORRECTLY SPECIFIED FOR XXXXXX. ASSUMED 1.' Read Section III, column 19.

MESSAGES (continued)

6. 'ERROR IN VARIABLE INTERACTION OR DEFINITION CARD FOR XXXXXX. REPORT SKIPPED.' Read 'SAMPLE SELECTION' in this manual.
7. 'SPECIFIED VARIABLE LENGTH FOR XXXXXX IN ERROR. REPORT SKIPPED.' Length of the variable as specified in columns 1-2 of the definition card is in error. Read 'SAMPLE SELECTION' in this manual.
8. 'THE AND/OR OPERATORS INCORRECT FOR XXXXXX. REPORT SKIPPED.' Read 'SAMPLE SELECTION' in this manual.

USER NOTE

When skill level is the 'average value' variable, the program bypasses cases if skill level is not 1, 3, 5, 7, or 9.

PROGRAM IDENTIFICATION

Name	AVGPCT (Average by % Performing)
Language	FORTRAN V
Date Written	November 1972/May 1973
Programmers	Weissmuller and Barton

FUNCTION

This program will calculate an average level (grade, skill level, etc.) which is based on the percent of members in each level actually performing each task. The tasks are listed with the resulting means. Options include:

1. Extracting skill level from a 5 digit AFSC,
2. Printing task titles in order of descending mean values,
3. Punching a deck of NTASK means in an (8X, 14F5.2) format,
4. Adding the Report to the REPORT file, and
5. Selecting a subsample from the tape using a 'SAMPLE SELECTION' procedure.

INPUT/OUTPUT

Input	a. HISTORY/KPATH file.
	b. Control cards.
Output	a. Task titles and means report.
	b. A deck of task means.
	c. The report added to the REPORT file.

CONTROL

Section I: FILENAMES CARD

Columns	
1- 6	'AVGPCT'
8-18	Filename of the HISTORY/KPATH file.
20-30	Filename of the REPORT file.

Section II: IDENTIFICATION

Columns	
1- 6	'AVGPCT'
8-15	Study ID on the HISTORY/KPATH file.

Section III: CONTROL CARD

Columns	
5	Variable code: 'A' if skill level must be extracted from 5 digit AFSC variable. 'S' if the variable is a single digit skill level. This will reject values = 2, 4, 6, 8 Blank will use a single digit and reject values not in the range (1 to 9). Use blank for average grade.
7-10	Variable-ID (Vxxx) of the variable to be averaged. This must be a background variable.
12	Number of PAGE HEADING cards (0, 1, 2, or 3)
14	'1' = Print task titles in descending sequence. '0' = Print titles in duty-task order.
16	'1' = Punch a means deck. See General Description for format. '0' = Do not punch the deck.
18-65	REPORT TITLE.
67	'0' = Add the report to the REPORT file. '1' = Do not save the report.

69-74	REPORT-ID.

Section IV: PAGE HEADING CARDS (optional, max = 3)

Columns	
1-80	Any alphanumeric information desired.

Section V: SAMPLE SELECTION CARDS

NOTE: This section is optional and may be deleted. If it is not present, all cases on the tape will be used. If a sample is desired, see 'SAMPLE SELECTION' in this manual for a complete description of sample selection procedures.

Section VI: END CARD

Columns	
1- 3	END

NOTE: Sections III through VI (control through END cards) may be repeated for stacked runs.

GENERAL DESCRIPTION

The program reads and rosters the control cards. Any errors will cause that set to be ignored. The program will halt upon encountering an end of file on the card. If there are no errors on the control cards, the input file is read. On the first pass the Study-ID is checked, relative location tables, etc., are read in and stored. On subsequent passes, these records are bypassed. The case data are scanned, cases are used or rejected if a sample selection is used. The level and task information is distributed as the file is being read. At the end of the case data, the means and standard deviations are computed for all tasks. On the first pass, the remainder of the tape is copied to disk. If requested, a deck of mean values is punched in an 8X, 14F5.2 format, where the 8X contains the following control information: cc 1-2 = card number, cc 3-4 = 'AP' for AVERAGE PERCENT, cc 5-8 = Study-ID from cc 7-10 of the IDENTIFICATION card. The task means are stored into descending order if required. The task titles, number of members in each level, means, and standard deviations are printed in the desired sequence and optionally copied to the REPORT file. The program resets and recycles if multi-reports are requested.

MESSAGES

1. 'SEARCHING FOR END CARD DUE TO ABOVE ERROR.' Self-explanatory.
2. 'AVGPCT CAN AVERAGE ONLY V VARIABLES - SEE CC7.' The control card specified a computed variable to be averaged, or the "V" was mispunched.
3. 'ABOVE CARD HAS INVALID VARIABLE ID IN CC8-10.' The variable ID must contain 3 numeric digits.
4. 'NUMBER OF VARIABLES IS NOT IN RANGE (1-9).' See cc2 of the Variable Interaction Card.
5. 'INCORRECT VARIABLE ID OR OPERATOR IN CC4-80.' See 'SAMPLE SELECTION' in this manual following program writeups.
6. 'VARIABLE ID DOES NOT MATCH CORRESPONDING ID.' The Variable Limits cards must be input in the same sequence as the variables appear on the Interaction Card.
7. 'INVALID INTEGER ABOVE - SEE CC9-14 + CC15-20.' C001, C002, and C003 are integer values and must have right-adjusted numeric limits.
8. 'UNEXPECTED END CARD - LIMITS CARD MISSING.' There must be exactly one limits card for each variable counted in cc2 of the Interaction Card.

9. 'END CARD IS MISSING - EXTRA CARDS PRESENT.' See above message.

10. 'UNEXPECTED END OF FILE ON CARD READER.' The last END card is missing.

11. 'END OF AVGPCT ON SSSSSSSS X OF Y RUNS WERE GOOD.'

    SSSSSSSS is the Study ID

    X is the number of completed reports

    Y is the number of attempted reports

    This means all processing is complete.

12. 'TAPE ID ON CARD DOES NOT MATCH TAPE ID.' Check ID in above message against the requested ID on the IDENTIFICATION CARD.

13. 'NO VALID CASES FOUND IN THE SAMPLE ON TAPE.' The sample selection procedure rejected all cases. The report was bypassed.

14. 'FIRST EOD NOT FOUND...ASSUME NCASE IS INCORRECT.' The NCASE in the Communications Region of the tape is not accurate. The run aborts.

USER NOTES

The means and standard deviations printed by this program use the percent of level members performing as the frequency. The percent performing is given by the actual number performing divided by the total number in the level, and multiplying the quotient times 100. Specifically

If:

For each level (or value of the variable)

TOT = Total number of members

PRF = Number performing a given task

PCTPRF = % performing = PRF/TOT * 100

For each task

F = Sums of PCTPRF for all levels

X = The value of the level (may be grade, skill level, etc.)

SUMFX = Sum of PCTPRF * X for all levels

SFXSQ = Sum of PCTPRF * X * X for all levels

Then:

Task mean = T = SUMFX/F

Task S.D. = Square root of (SFXSQ/F - T * T)

PROGRAM IDENTIFICATION

Name	CORREG (Correlation and Regression)
Language	FORTRAN V
Date Written	Jan 1972/May 1973
Programmer	Stacey

FUNCTION

This program package extracts up to 100 computed and background variables from a CODAP KPATH or HISTORY file and computes correlation matrices and regression problems. The correlation program computes and prints the correlation matrix, number of valid and invalid cases in the sample, and means and standard deviations of variables. These same computations are also put on file for future reference, and they also remain in core if they are to be fed into the regression program immediately. A subsample may be selected for correlation according to the value (0 or 1) of a computed variable. In this routine, a series of regression problems is computed iteratively. Beginning with the best combination of three predictors at iteration 1, the best remaining combination of three predictors is added to the model at each subsequent iteration. The "best remaining combination" is that which adds the most to the value of R^2 when used in conjunction with all combinations of predictors selected at previous iterations. Predictors may be used more than once during the iterative process. Iterations are continued until the increase in the value of R^2 over the previous iteration is less than some amount specified by the requester. Variable ID's, R^2 values, and the error sum of squares are reported for each iteration. The standard and raw score weights for each variable as they exist upon completion of the final iteration are reported, as well as the regression constant.

INPUT/OUTPUT

Input	a. HISTORY file or KPATH file.
	b. Control cards.
Output	a. CORRELATION MATRIX file.
	b. Printed matrix and regression output.

CONTROL

Section I: FILENAMES CARD

Columns	
1- 6	"CORREG"
8-18	Filename of the HISTORY or KPATH file, or, Filename of the input CORRELATION MATRIX file.
20-30	Filename of the output CORRELATION MATRIX file.

NOTE: When correlations and regression, or correlations alone are computed, the Filename of the HISTORY or KPATH file is given in field 8-18 and the Filename of the output MATRIX file is given in field 20-30.

When regression problems alone are computed, the Filename of the input MATRIX file is given in field 8-18 and field 20-30 is blank.

Section II: IDENTIFICATION

Columns	
1- 6	"CORREG"
8-15	Study I.D. on the HISTORY or KPATH file if mounted. (This file is not used when regression problems alone are computed, in which case this field would be blank.)
17	Number of PAGE HEADING CARDS following. May be any number 0 thru 9.

Section III: PAGE HEADING CARDS

Columns	
1-80	Any information the user desires. PAGE HEADING CARDS are not required. A maximum of 9 cards may be used.

Section IV: CORRELATION HEADER (Sections IV and V are not used if a CORRELATION MATRIX already exists and is input to this run.)

Columns	
1- 4	"CORR"
5- 7	The number of variables punched in the cards that follow. Maximum is 100.
9-16	CORRELATION MATRIX ID. This field will be written on the CORRELATION MATRIX output file.
18-21	Subsample selector variable. If the total sample is desired, leave this field blank. If specified, this variable must be a computed variable (Cxxx) which has been set to 0 or 1 by a prior run.

23	If a selector variable has been specified, this field must equal 0 or 1. Specify 0 to select cases only when the selector variable equals 0 and specify 1 to select cases only when the selector variable equals 1.

Section V: VARIABLE SPECIFICATION CARD(S)

Columns	
1- 4	1st variable
6- 9	2nd variable
76-79	16th variable

Variables are punched 16 to a card for as many cards as required. Both Computed variables (Cxxx) and Background variables (Vxxx) may be specified. Variables may be punched in any order; however, the program will reorder them low to high, Computed variables followed by Background variables. Background variables cannot exceed 6 characters (1A6) in size. A maximum of 100 variables combined (Cxxx and Vxxx) is allowed. C001, C002, C003 cannot be used.

NOTE: The run may terminate at this point if a CORRELATION MATRIX only is desired. If regression problems are desired, continue with the "REGR" card as follows.

Section VI: REGRESSION HEADER

Columns	
1- 4	"REGR"
7	Iteration print option. Punch "1" to include a print of the iteration process along with the regression output. Leave blank otherwise.

9-16	CORRELATION MATRIX file ID. This field will be compared to the ID from the input CORRELATION MATRIX file.

Section VII: REGRESSION PROBLEM DEFINITION CONTROL CARD(S)

Columns	
1-10	Problem identification. This field will be printed with regression output.
11-18	Stop criterion punched with a decimal point. Whenever the difference in the RSQ values for two successive iterations is less than this given value, the iteration process will stop.
19-20	Maximum number of iterations. If this field contains a positive number, the iteration process will stop whenever the number of iterations performed equals the value in this field or whenever the stop criterion is met, whichever occurs first. If this field is left blank, only the stop criterion will be considered.
21-23	Number of 3-digit fields following. If the number of fields is greater than 19, field 20 and over must be punched in columns 24-80 of additional control cards. This field will always contain an odd number since predictor variables must be listed in pairs indicating a range, and the criterion variable is listed as a single 3-digit field.

24-80	A sequence of 3-digit fields of the form N1, N2, N3, ..., NX, NY, NC, where these are integer constants. This will cause variables N1 thru N2, N3 thru N4, N5 thru N6, ..., NX thru NY to be included as predictor variables, and variable NC to be included as the criterion. The criterion variable is always listed last.

Section VIII: F-RATIO CONTROL CARD(S)

Columns	
1-10	Problem identification.
21-23	"008". Eight identifies an F-Ratio control card.
24-26	RSQ number of the full model.
27-29	RSQ number of the restricted model. The RSQ numbers for the full and restricted models correspond to the order in which the regression control cards were entered.
30-35	Degrees of freedom in the numerator.
36-41	Number of independent vectors in the full model.
42-44	Variable number of the criterion variable. This field is used if the sum of squares between the full and restricted models is desired, otherwise leave blank.

NOTE: The degrees of freedom in the denominator is equal to the number of valid cases minus the number in cc 36-41.

GENERAL DESCRIPTION

Control cards are read. If a "CORR" card was encountered, the HISTORY or KPATH file is read and the study I.D. is checked. The variable request card is read and the requested variable numbers are sorted low to high, computed variables first, followed by background variables. Variable definitions are read from the HISTORY or KPATH file and the requested variable definitions are printed. Case data records are read from the HISTORY or KPATH file and requested variables are extracted and checked. If any one of the requested variables is invalid (non-numeric, blank, or greater than 6 characters in size), the case data record is bypassed. If all variables are valid, they are written in a format acceptable to the PERSUB CORRLB routine. The CORRLB routine is called and a CORRELATION MATRIX is computed. The matrix is written on tape for future regression problems. If no control cards follow, the program terminates. If a "REGR" card follows, the program calls the PERSUB REGREF routine to compute regression problems for all control cards following. If the program had read a "REGR" card instead of a "CORR" card as described above, the assumption would have been that a CORRELATION MATRIX was already available, and the program would go directly into the regression phase.

MESSAGES

1. "STUDY ID FROM HISTORY OR KPATH FILE DOES NOT MATCH CONTROL CARD. FILE ID IS XXXXXXXX." Self-explanatory.
2. "HISTORY VARIABLE XXX IS NOT BETWEEN 1 AND 6 CHARACTERS ACCORDING TO THE VARIABLE DICTIONARY ENTRY. RUN TERMINATED." Either the format that is specified on the background variable definition card (on the HISTORY or KPATH file) is wrong or the variable is greater than 6 characters in size.
3. "ALL CASE DATA REJECTED. A BACKGROUND VARIABLE CONTAINING ONLY ALPHA DATA MAY HAVE BEEN REQUESTED." Self-explanatory.
4. "BACKGROUND VARIABLE XXX DOES NOT EXIST ON THE CODAP FILE." Self-explanatory.

5. "THE NUMBER OF VARIABLES SPECIFIED ON THE CORRELATION HEADER CARD IS OUT OF RANGE." Columns 5-7 must contain a number 1 thru 100.
6. "COMPUTED VARIABLE XXX DOES NOT EXIST ON THE CODAP FILE." Self-explanatory.
7. "VARIABLE NUMBER XXX FROM THE VARIABLE SPECIFICATION CARD(S) DOES NOT BEGIN WITH V OR C." All variables must begin with V or C.
8. "CORRELATION MATRIX ID NO MATCH. TAPE IS XXXXXXXX. CARD IS XXXXXXXX." The CORRELATION MATRIX ID from columns 9-16 of the "REGRESSION HEADER CARD" does not match the matrix ID found on the CORRELATION MATRIX file.
9. "REGRESSION CONTROL CARD CONTAINS AN INVALID VARIABLE. PROB. ID = XXXXXXXXXX." A variable was read which does not exist in the CORRELATION MATRIX. Possible mispunched card.

PROGRAM IDENTIFICATION

Name	DIAGRM (Diagram)
Language	FORTRAN V
Date Written	1970/May 1973
Programmer	CSC/Weissmuller

FUNCTION

This program uses data from the GRMBRS program to generate a treelike diagram that visually displays the order in which groups merged during the hierarchical grouping process. Each group is represented by a rectangular block of data containing the same information found in GRMBRS. Rows and columns of asterisks show the branches leading from a group to its subgroups. Control cards can be used to limit the number and type of groups displayed by DIAGRM.

INPUT/OUTPUT

Input	GRMBRS output file
Output	Grouping actions shown graphically on printed output.

CONTROL

Section I FILENAMES CARD

Columns	
1-6	'DIAGRM'
8-18	Filename of the GRMBRS output file.

Section II CONTROL CARD

Columns	
1-6	'DIAGRM'
8-15	Study Identification (alphanumeric)
17	Number of PAGE HEADING cards to follow. 0-3 cards are allowed.

CONTROL (continued)

Section III

Columns	
1-80	Any information desired.

Section IV DEFINITION CARD

Columns	
1-5	X = an average between group value used as the starting point for selecting starter groups. This value will vary depending upon the average between at stage N/V. If the average at stage N/V is $\leq$ the input X, let X = the input X. If this value is > the input value of X, let X = the observed value.
6-10	Y = an average between group value used as the lower limit cutoff in selecting starter groups. (A starter group is one of the beginning groups for the hierarchical grouping actions to be diagrammed)
11-15	W = an integer value that defines the minimum size of a group to be selected as a starter group. Must be right adjusted.
16-20	N/V where N is the number of stages in the job analysis to be diagrammed. Usually V = 2 so that N/V = N/2 for the computation of the average between value to compare to the input X value. N/V is an integer that must be right adjusted in its field.

CONTROL (continued)

Section IV DEFINITION CARD (continued)

Columns	
22	If the minimize collapse option was used in the group program, this column must be "1". If the maximize option was used, leave blank.

X and Y are floating point values that must have a decimal point punched in the field.

GENERAL DESCRIPTION

DIAGRM reads the control and title cards to establish values for X, Y, W and N/V. Then the input file from GRMBRS is read to establish the study identification and the number of stages in the grouping action. To select the starting groups begin at between group value X in the GRMBRS file proceeding sequentially toward stage N-1, selecting all groups ≥ W in size which have independent KPATH ranges. This process continues until stage N-1 is reached. The program then goes back to the stage immediately preceding between group value X on the GRMBRS file and proceeds sequentially toward stage 1, selecting all groups ≥ W in size which have KPATH ranges independent of groups already selected, until between group value Y is reached.

The starter groups plus all groups formed at subsequent grouping stages down to stage 1 are computed and presented in the computerized diagram. The following data items are computed and printed for every group presented in the computerized diagram. (Line numbers refer to the layout of the data item) Line 1 - stage number and number of members in the group; Line 2 - KPATH range; Line 3 - average between and within group values rounded to the nearest tenth.

MESSAGES

1. INCORRECT STUDY ID XXXXXXXX DOES NOT MATCH TAPE STUDY ID XXXXXXXX PROGRAM ABORT. Study identification on control card and GRMBRS file fail to match.
2. NUMBER OF TITLE CARDS CANNOT EXCEED 3. PROGRAM ABORT. Self explanatory.
3. NO AVE BETWEEN $\geq$ X ON INPUT TAPE. Program aborted. Check file assignments.
4. NO STARTER GROUPS FOUND. Program aborted. Check file assignments and DEFINITION card for X, Y, and W.
5. TOO MANY STARTER GROUPS FOUND, CHANGE INPUT AND RESUBMIT. Program aborted. Probably need to increase W to stay under maximum of 200 starter groups.

PROGRAM NOTES

1. A maximum of 200 starter groups is allowed.
2. NSTAGE must be less than 2,000.

PROGRAM IDENTIFICATION

Name	DIST2X (Two-way Distribution)
Language	FORTRAN V
Date Written	1972/May 1973
Programmer	Weissmuller

FUNCTION

This program prints a distribution of two variables from a HISTORY/KPATH file. Samples may be selected by specifying a dichotomous (categorical membership) variable. User may specify to select on 1 or 0. Options include calculation and printing of:

1. Means and standard deviations
2. Percentage of row, column, and/or total frequency represented in each cell, and
3. An interval labelled OTHER which contains all data not within the LOW and HIGH values given. This option causes all numeric data to be used in means and standard deviations.

Any combination of these options may be used on rows and/or columns independently of the other.

INPUT/OUTPUT

Input	a. HISTORY/KPATH file.
Output	a. Printed distribution and summary.

CONTROL

Section I FILENAMES CARD

Columns	
1- 6	DIST2X
8-18	Filename of the HISTORY/KPATH file.

Section II IDENTIFICATION

Columns	
1- 6	DIST2X
8-15	Study-ID
17	Number of PAGE HEADING cards following. 0 to 3 cards are allowed.
19-22	NOGO = Terminate run after scanning card input for errors. Blank = Do not terminate the run after the card scan.
24	1 = Print TOT PERCENT which indicates the percentage of the total frequency represented in each cell. Blank = Do not print TOT PERCENT.

Section III PAGE HEADING CARDS

Columns	
1-80	Any alphanumeric headings desired.

Section IV SAMPLE RESTRICTION (All cases used when this card is not present,)

Columns	
1- 6	SAMPLE
8-11	Variable ID on which to select variable. May be either a history or computed variable.
13	1 = Select a case only if the value of the above variable is 1. 0 = Select a case only if the value of the above variable is 0. NOTE: If the variable is a history variable any character string may be used for exact match selection.

Section V ROW DEFINITION (1 required for each distribution)

Columns

1- 3 ROW

8-11 Variable ID. Note that C001, C002, C003 and history variables with more than 6 characters may not be used.

NOTE: Use cc13-18, 20-25, 27-32 for history variables only.

13-18* LOW = lowest acceptable history variable value.

20-25* HIGH = highest acceptable history variable value.

27-32* SIZE =

a. If blank, the LOW and HIGH will represent one alphanumeric interval and will be printed as a single row. Additional alphanumeric rows may be specified using continuation cards. (See Section VII)

b. If 0, the LOW and HIGH will represent one numeric interval and will be printed as a single row. Additional numeric rows may be specified using continuation cards. (See Section VII)

c. If an unsigned integer, (value = N), the LOW and HIGH will represent the absolute minimum and maximum values for all rows to be printed. The range from LOW to HIGH will be divided into smaller intervals of size N. The last interval may be smaller than size N since it will terminate with the value HIGH. If N is greater than HIGH - LOW, a single row will be written for this entire distribution. This usage does not allow continuation cards.

*These fields are used for history variables only. If either LOW or HIGH is left blank, a row will be written for each unique alphanumeric value found, and SIZE will be ignored. If not left blank, each field must contain the number of digits or characters specified in the format of the dictionary entry for this history variable. The values must be left justified with leading zeroes for numerics with less than the maximum number of digits. (e.g., Format = A3, 001 is written in cc13-15.)

34 1 = Print the mean and standard deviation of the valid column values in each row. These statistics are printed at the end of each row, down the right-hand side of the page. The means and standard deviations are based on data which is numeric and within the LOW to HIGH range specified on the COLUMN DEFINITION card. If no range is specified or cc36 = 1 on the COLUMN DEFINITION card, then all numeric data, regardless of range, is used in these calculations.

Blank = Do not print these statistics.

35 1 = Print the ROW PERCENT which indicates the percentage of the row represented by each cell.

Blank = Do not print ROW PERCENT.

36 1 = Use all numeric data, regardless of range, in computing means and standard deviations. All data not in the LOW to HIGH range will be distributed into a row labelled OTHER. Note that this option affects the mean row value printed at the bottom of the page.

Blank = Use only numeric data between LOW and HIGH and consider other data as invalid.

NOTE: Use cc40-49, 50-59, 60-69 for computed variables only.

40-49** LOW = lowest acceptable computed variable value.

50-59** HIGH = highest acceptable computed variable value.

60-69** SIZE =

a. If blank or 0.0, the LOW and HIGH will represent one numeric interval which will be printed as a single row. Additional numeric rows may be specified using continuation cards. (See Section VII)

b. If an unsigned decimal number, (value = N.d), the LOW and HIGH will represent the absolute minimum and maximum values for all rows to be printed. The range from LOW to HIGH will be divided into smaller intervals of the size N.d. Intervals include low limits, but not the high. The last interval is special since it terminates with the value HIGH. Hence it may be shorter than N.d and it includes its upper limit. If N.d is greater than HIGH - LOW, a single row will be printed. This usage does not allow continuation cards.

**These fields are used for computed variables only. Any field left blank is assumed to be 0.0. The values in LOW, HIGH, and SIZE must either contain a decimal point or be right justified.

NOTE: Continuation cards for the rows must appear here.

Section VI COLUMN DEFINITION (1 required for each distribution. 450 columns maximum.)

Columns	
1- 6	COLUMN
8-11	Variable ID. Note that C001, C002, C003, and history variables with more than 6 characters may not be used.

NOTE: Use cc13-18, 20-25, 27-32 for history variables only.

13-18*	LOW = lowest acceptable history variable value.
20-25*	HIGH = highest acceptable history variable value.
27-32*	SIZE =

a. If blank, the LOW and HIGH will represent one alphanumeric interval and will be printed as a single column. Additional alphanumeric columns may be specified using continuation cards. (See Section VII)

b. If 0, the LOW and HIGH will represent one numeric interval and will be printed as a single column. Additional numeric columns may be specified using continuation cards. (See Section VII)

c. If an unsigned integer, (value = N), the LOW and HIGH will represent the absolute minimum and maximum values for all columns to be printed. The range from LOW to HIGH will be divided into smaller intervals of size N. The last interval may be smaller than size N since it will terminate with the value HIGH. If N is greater than HIGH - LOW, a single column will be written for this distribution. This usage does not allow continuation cards.

*Same restrictions as ROW DEFINITION Card.

34 1 = Print the mean and standard deviation of the valid row values in each column. These statistics are printed at the bottom of each column, along the bottom of the page. The means and standard deviations are based on data which is numeric and within the LOW to HIGH range specified on the ROW DEFINITION card. If no range is specified, or if cc36 = 1 on the ROW DEFINITION card, then all numeric data, regardless of range, is used in these calculations.

Blank = Do not print these statistics.

35 1 = Print the COL PERCENT which indicates the percentage of the column represented by each cell.

Blank = Do not print COL PERCENT.

36 1 = Use all numeric data, regardless of range, in computing means and standard deviations. All data which is not in the LOW to HIGH range will be distributed into a column labelled OTHER. Note that this option affects the mean column value printed on the right hand side of the page.

Blank = Use only numeric data between LOW and HIGH and consider all other data as invalid.

NOTE: Use cc40-49, 50-59, 60-69 for computed variables only.

40-49** LOW = lowest acceptable computed variable value.

50-59** HIGH = highest acceptable computed variable value.

60-69** SIZE =

a. If blank or 0.0, the LOW and HIGH will represent one numeric interval which will be printed as a single column. Additional numeric columns may be specified using continuation cards. (See Section VII)

b. If an unsigned decimal number, (value = N.d), the LOW and HIGH will represent the absolute minimum and maximum values for all columns to be printed. The range from LOW to HIGH will be divided into smaller intervals of size N.d. Intervals include low limits, but not the high. The last interval is special since it terminates with the value HIGH. Hence, it may be shorter than N.d and it includes its upper limit. If N.d is greater than HIGH - LOW, a single column will be printed. This usage does not allow continuation cards.

**Same restrictions as for ROW DEFINITION Card.

Section VII CONTINUATION CARDS

If used, continuation cards must follow the DEFINITION CARD to which they refer. These cards must be input in ascending scientific sequence ordered on the LOW field. No interval may overlap with another.

Columns	
13-18*	LOW = lowest acceptable history variable value.
20-25*	HIGH = highest acceptable history variable value.
40-49**	LOW = lowest acceptable computed variable value.

50-59** HIGH = highest acceptable computed variable value.

*Same restrictions as ROW DEFINITION Card.

**Same restrictions as ROW DEFINITION Card.

Section VIII END

Columns

1- 3 END

NOTE: Sections II - VIII (DIST2X through END) may be repeated for stacked runs.

PROGRAM NOTES

1. No more than 450 columns can be generated.
2. If 225 to 450 columns are used, no more than 800 rows may be generated.
3. No more than 224 continuation cards may follow a DEFINITION Card.
4. If a DEFINITION Card indicates a numeric range, continuation cards, if used, must contain all numerics or an error halts execution.
5. Continuation cards, if used, must be input in ascending scientific sequence on LOW and no intervals may overlap.
6. The DEFINITION Card determines which LOW - HIGH - SIZE set to use. If cc8 is V, use fields beginning in cc13. If cc8 is C, use fields beginning in cc40. The options remain in the same columns regardless of the type of variable.
7. Fields marked with * or ** must conform to the restrictions given in the ROW DEFINITION card, Section V.
8. In stacked runs, each distribution must begin with a DIST2X card and terminate with an END card.
9. The number of heading cards must agree with the number in cc17 of the DIST2X card, and be 3 or less.

10. The variable specified on the SAMPLE card must be a categorical membership variable. It is checked against a 1 or 0 only.
11. Computed variables C001, C002, and C003 may not be used.
12. Only history variables of 6 or less characters may be used.
13. A 1 in card column 36 of a ROW DEFINITION card causes an additional, OTHER, row to be printed. Similarly, on the COLUMN DEFINITION card, it causes an additional, OTHER, column to be printed. All values listed under the OTHER headings represent data which is invalid or not within the specified range. If printed, the valid, but out of range, numeric data is used in computing the means and standard deviations.
14. IF:
    A. An OTHER option is used,
    B. An alphabetic range is specified, or
    C. A raw distribution is requested, then

    ROW and COLUMN percents will be based on the respective TOTAL, and the TOT percents will be based on the TOTAL-TOTAL or GRAND TOTAL figure.

    If none of the above apply, then the ROW percent is based on the valid for each row and the COLUMN percent is based on the valid for each column. The TOT percent is based on the VALID-VALID or, the total number actually distributed.
15. Means and Standard Deviations are based on the distributed data which precedes it. The only exceptions are the mean and standard deviation in the VALID row or column. These statistics are based on all valid data, regardless of whether or not it has been distributed. It yields the same result which would be achieved by ignoring the other variables range limits.

GENERAL DESCRIPTION

The program reads in the DIST2X card, checks program ID and records other information. If cc17 is not blank or 0, Heading cards are read in and saved. The next card is read to see if it is a valid control word (SAMPLE, COLUMN, ROW, END, or blank). If it isn't, the card is ignored. The cards are processed and errors are flagged. On a normal run, a single error will cause immediate termination. For a NOGO, all cards are checked for errors. If errors are detected or a NOGO was specified, the program terminates or skips to the next distribution if runs are stacked.

The input file is rewound and the Communications Region is read. The Study-ID is then checked. A mismatch causes an error message and skips to next distribution. The Relative Location Table is read and the location of required history variables is stored. The Dictionary is searched for the requested variables and the formats are decoded. If no Dictionary entry for a requested variable is found, the program skips to the next distribution.

The cases are then read in sequentially. After a case is read, its row and column values are decoded if necessary. If intervals are used, they are determined for the row then the column. The row - column pairs are compared with a table in core. If it has been used before, the statistical information is added to the matching entry. If it has no match in the table, it is added. If the table overflows, it is merged to disk, and the core table is considered empty. After that, the statistical information for the columns is added into the proper column. There are only 450 column locations reserved and if this table overflows, the program skips to the next distribution.

Once all cases have been processed, the input file is rewound. The remaining case data in core is merged to disk and disk is rewound. The disk is read back, accumulating information into single rows and writing these rows out to the alternate disk unit. The alternate disk is rewound. The rows are read back from the alternate disk unit. If an entire row with options requires more than 10 columns, the disk unit must be rewound for each multiple of 10 columns in order to print the distribution. Once all the rows have been printed, a summary is organized and printed. The program then skips to the next distribution.

If there are no more distributions in the stack, processing ends.

ERROR MESSAGES

1. 'REQUESTED TAPE ID = XXXXXXXX, TAPE MOUNTED = YYYYYYY.' The Tape ID did not match.
2. 'PROGRAM IS LIMITED TO 3 HEADING CARDS ... CARD IGNORED.' CC17 of the DIST2X Card must be 3 or less. Processing continues, ignoring all cards in excess of 3.
3. 'INCORRECT PROGRAM ID.' The Program ID is DIST2X and it must appear in the first 6 columns of the DIST2X Card.
4. 'ROW CARD IS MISSING OR MISPUNCHED.' Exactly 1 ROW card is required for each distribution. ROWƀƀƀ must be in the first 6 columns of the ROW DEFINITION Card.
5. 'COLUMN CARD IS MISSING OR MISPUNCHED.' Exactly 1 COLUMN Card is required for each distribution. COLUMN must be in the first 6 columns of the COLUMN DEFINITION Card.
6. 'NUMBER OF INTERVALS EXCEED PROGRAM LIMITS.' See RESTRICTIONS 1 through 3 above.

7. 'NON-NUMERIC DATA IN ABOVE FIELD.' This message occurs for history variables only. If the SIZE field was an unsigned numeric (0, 1, 2,...) the LOW and HIGH must also be numeric and contain the same number of digits. (If the HIGH = 999, then LOW and SIZE must contain 3 digits, 001 and 005 for example. These 3-digit fields must be left justified in HIGH, LOW, and SIZE.)

8. 'SAMPLE VARIABLE NOT IN DICTIONARY.' or
   'COLUMN " " " " .' or
   'ROW " " " " .'
   The variable requested was not found on the input file. If the tape-ID matched, it means that an older file from the same study was used. Locate the most current file which contains the above variables and re-run the distribution.

9. 'NOGO SPECIFIED - EXECUTION SUSPENDED.' All cards were checked for errors.

10. 'PROGRAM ABORTS - FOLLOWING CARDS IGNORED.' An error was detected in a normal run. All cards up through the next END card are ignored.

11. 'DUPLICATE ROW CARD - FATAL ERROR.' or 'DUPLICATE COLUMN - FATAL ERROR.' Each distribution must have exactly one ROW DEFINITION Card and exactly one COLUMN DEFINITION.

12. 'UNEXPECTED BLANK CONTROL WORD-IGNORED.' Columns 1-6 were blank and the program did not expect a continuation card. If SIZE is not blank or 0, continuation cards cannot be used.

13. UNRECOGNIZABLE CONTROL WORD - IGNORED.' Columns 1-6 did not contain a control word. If an END card is missing within a stacked run, this message will appear for the next DIST2X Card. Both distributions will be skipped.

14. 'DIST2X CANNOT USE MULTIPLE WORD VARIABLES.' The program cannot use history variables which require more than 1 word of storage.

15. 'INVALID VARIABLE ID (MUST BE C OR V).' CC8 of ROW, COLUMN, and SAMPLE Card must be either a C or a V.

16. 'END OF DIST2X.' All processing for this run is complete.

PROGRAM IDENTIFICATION

Name	DUVARS (Duty Variables)
Language	FORTRAN V
Date Written	Aug 1972/May 1973
Programmer	Stacey

FUNCTION

Computes duty values from a CODAP file (KPATH or HISTORY) using any one or all of the following 3 options: (Option 1) Compute percent time spent in each duty for each case; (Option 2) Compute the number of tasks performed in each duty for each case; (Option 3) Compute the percentage of total tasks performed in each duty for each case. Duty values computed from these options can be added to the CODAP file as new Vxxx variables if desired. A printed report is always provided and can be saved on the REPORT file if desired. An updated dictionary is printed if duty variables are added to the CODAP file.

INPUT/OUTPUT

Input
a. HISTORY or KPATH file.
b. Control cards.

Output
a. Updated HISTORY or KPATH file (optional).
b. REPORT file w/REQUEST card(s) (optional).
c. Printed report.

CONTROL

Section I: FILENAME CARD

Columns	
1- 6	'DUVARS'
8-18	Filename of the HISTORY/KPATH file.

20-30	Filename for the updated HISTORY/KPATH file.
32-42	Filename of the REPORT file.

Section II: CONTROL

Columns	
1- 6	'DUVARS'
8-15	Study I.D. on the KPATH/HISTORY file.
17(Opt 1)	'1' = Select Option 1 (compute and print report time spent by duty for all cases).
19(Opt 1)	Number of PAGE HEADING cards for the Option 1 report. 0-2 cards are allowed. This count does not include the Section III card.
21(Opt 1)	'1' = Add the Option 1 report to the REPORT file. '0' = Do not save the report.
23(Opt 1)	'1' = Add the Option 1 variables to the KPATH/HISTORY file as Vxxx type CODAP variables. A new variable will be created for each duty in the sample.
25(Opt 2)	'1' = Select Option 2 (compute and print the number of tasks performed by duty for all cases).
27(Opt 2)	Number of PAGE HEADING cards for the Option 2 report. 0-2 cards are allowed. This count does not include the Section III card.
29(Opt 2)	'1' = Add the Option 2 report to the REPORT file. '0' = Do not save the report.
31(Opt 2)	'1' = Add the Option 2 variables to the KPATH/HISTORY file as Vxxx type CODAP variables. A new variable will be created for each duty in the sample.

33(Opt 3)	'1' = Select Option 3 (compute and print the percent of total tasks performed by duty for all cases).
35(Opt 3)	Number of PAGE HEADING cards for the Option 3 report. 0-2 cards are allowed. This count does not include the Section III card.
37(Opt 3)	'1' = Add the Option 3 report to the REPORT file. '0' = Do not save the report.
39(Opt 3)	'1' = Add the Option 3 variables to the KPATH/HISTORY file as Vxxx type CODAP variables. A new variable will be created for each duty in the sample.

Section III: REPORT TITLE (Required)

Columns	
1-48	Any information the user desires. This field is printed on the page number line of each page of the report and is punched into the REQUEST card. One REPORT TITLE is required for each selected option.
50-55	Special I.D. of this report. This field is written on the REPORT file and punched into the REQUEST card if this report is added to the REPORT file. It is also printed on the page number line of each page of the report.

Section IV: PAGE HEADING CARD(S) (Optional)

Columns	
1-80	Any information the user desires. HEADING CARD information is printed on the first page of the report. 0-2 cards may be used.

NOTE: REPORT TITLE Cards (Section III) and PAGE HEADING Cards (Section IV) must be together as a set for each option requested. Option 1

REPORT TITLE and HEADING CARDS follow the Section II Control Card, followed by Option 2 REPORT TITLE and HEADING CARDS, followed by Option 3 REPORT TITLE and HEADING CARDS. Cards for options not requested would be omitted.

GENERAL DESCRIPTION

The Communication Region from the KPATH/HISTORY file is read and the study I.D. is compared to the control card study I.D. Control card options are range checked and the REPORT TITLE and PAGE HEADING cards are read and saved for later printing. If duty values are to be added to the KPATH/HISTORY file, the Communication Region and Relative Location table and the Variable Dictionary are updated. Case data is read from the KPATH/HISTORY file and all 3 duty options are computed and written on a scratch unit for later printing. Duty values are added to the KPATH/HISTORY file as background variables (whole numbers - A3 format) according to the options requested by the control card. When duty values have been computed for all cases, an updated dictionary of variables is printed, followed by a printed report containing case control number, number of tasks performed, and "NDUTY" duty values for each case in the sample. The report is added to the REPORT file if requested. Each requested option will generate one separate, independent report.

NOTE: This program assumes that Duty Titles are A, B, C, D, etc., and does not search the end of the KPATH/HISTORY file for that information.

MESSAGES

1. 'STUDY ID NO MATCH. TAPE ID IS XXXXXXXX.' The Study ID from the KPATH/HISTORY file does not match Study ID from the control card.

2. 'COLUMN 17 OR 25 OR 33 OF THE DUVARS CONTROL CARD IS IN ERROR. RUN TERMINATED.' Read Section II of the program write-up.
3. 'COLUMN 19 OR 27 OR 35 OF THE DUVARS CONTROL CARD IS IN ERROR. RUN TERMINATED.' Read Section II of the program write-up.
4. 'COLUMN 21 OR 29 OR 37 OF THE DUVARS CONTROL CARD IS IN ERROR. RUN TERMINATED.' Read Section II of the program write-up.
5. 'COLUMN 23 OR 31 OR 39 OF THE DUVARS CONTROL CARD IS IN ERROR. RUN TERMINATED.' Read Section II of the program write-up.
6. 'THE MAXIMUM NUMBER OF WORDS OF COMPUTED AND BACKGROUND VARIABLES ALLOWABLE IS XXXX. THIS RUN WILL CREATE XXXX.' The addition of duty variables (one variable added for each duty) to the KPATH/HISTORY file has exceeded the CODAP limit. Re-run with fewer options or run without adding any variables to the KPATH-HISTORY file.
7. 'THE LAST VARIABLE DICTIONARY ENTRY FROM THE CODAP TAPE IS ERRONEOUS IN COLUMNS 1-6. THEY ARE XXXXXX.' These columns must be of the form 'Vxxxb=.' The KPATH/HISTORY file must be re-created.

PROGRAM IDENTIFICATION

Name	EXTRCT (Extracting, Reordering and Printing Reports)
Language	FORTRAN V
Date Written	1966/Aug 1973
Programmer	CSC/Weissmuller

FUNCTION

To extract reports selected by REQUEST cards from a REPORT file, and create a new REPORT file reordered in the sequence of the input REQUEST cards. A new table of contents is produced and a new set of REQUEST cards is punched.

INPUT/OUTPUT

Input
- a. REPORT file.

Output
- a. Optionally, a new REPORT file in order of input REQUEST cards.
- b. New REQUEST cards for the new REPORT file.
- c. Printed report.

CONTROL

Section I FILENAMES CARD

Columns	
1- 6	EXTRCT
8-18	Filename of the REPORT file.
20-30	Filename for the new REPORT file. (EDIT/PRINT OPTION ONLY)

Section II CONTROL CARD

Columns	
1- 6	EXTRCT

8	blank = EDIT/PRINT. Create a new REPORT file and punch new REQUEST cards for the selected reports. Print the requested reports with a table of contents. 1 = PRINT. Read the REPORT file and print all reports and a table of contents. No new REPORT file is created and no REQUEST cards are punched. For this option, Section III is omitted. 2 = INDEX. For each report, print only Report ID, Report Title, and Report number. Section III is omitted.
10-13	STUDY NUMBER to be punched in request cards. (EDIT/PRINT only)
15	Specifies the number of complete print passes to be made (1 to 9). The REPORT file rewinds for each pass. Multiple additional copies are obtained by using multiple part paper. Two passes on two part paper yields 4 copies. If blank, program assumes 1.
17-18	Specifies the number of pages to print of each report. Zero or blank prints full report.
20-22	Applies only to the PRINT option and specifies the page number at which to start a print recovery. If a machine error occurs during the print phase, this option allows the user to skip pages which are already printed. If multiple copies are requested, only the first copy will be partial; all subsequent copies will begin with page 1.

Section III REQUEST CARDS

This section must be input when the EDIT/PRINT option is used. The user may limit the number of pages to print of each requested report by punching the desired limit in columns 75-76. This overrides the maximum page count given in cc 17-18 of the control card for this report only. A maximum of 700 REQUEST cards is allowed.

GENERAL DESCRIPTION

The program reads the Filenames card and initializes the files. It then reads the control card and determines what type of Extract run is requested. If it is an Index run, the report file is read and the Report ID, Report Title, and Report number from each report is printed. The run is then terminated. If EDIT/PRINT is requested, REQUEST cards are read and the Report numbers and page limit numbers are saved. Reports are read from the REPORT file and those reports matching the report numbers are written to disk. Reports are then printed from disk in the sequence of the Report numbers. Report length is controlled by cc 17-18 of the CONTROL card unless changed by cc 75-76 of the REQUEST cards. Pages are renumbered 1-N, a REQUEST card with a new Report number is punched for each report, each report is written on the new REPORT file, and a table of contents is printed at the end of all reports. The run is then terminated unless multiple passes are requested. If the PRINT option is requested, the entire REPORT file is printed. Pages are renumbered 1-N, and a table of contents is printed at the end of all reports. If page limit is specified, only that number of pages is printed for each report. If multiple printer passes is requested, the program recycles. If not, the program terminates.

MESSAGES

1. 'INVALID NUMERIC ABOVE.' Entries in card columns 15, 17-18, and 20-22 must be right adjusted numeric values.
2. 'THIS RUN EXCEEDS THE 700 REQUEST CARD LIMIT. XXX CARDS WERE READ.' Self explanatory.
3. 'ENCOUNTERED END OF REPORT FILE AFTER REPORT XXXX WHILE SEARCHING FOR REPORT YYYY.' A report was requested which cannot be located on the REPORT file.
4. 'THE RUN EXPECTED REQUEST CARDS BUT NONE WERE FOUND.' When the EDIT/PRINT option is used (cc 8 = blank), REQUEST cards must be input.
5. 'RUN HALTS DUE TO ABOVE ERROR.' Self explanatory.
6. 'INVALID PAGE MAX IN CC 75-76.' Self explanatory.
7. 'INVALID REPORT NUMBER IN CC 78-80.' Self explanatory.

PROGRAM IDENTIFICATION

Name	GRMBRS (group membership)
Language	FORTRAN V
Date Written	1966/May 1973
Programmer	CSC/Weissmuller

FUNCTION

This program produces an information report that identifies the two groups combining at each stage of the hierarchical grouping process. The information includes the stage number, the number of members in the combined group, the number of members in each of the combining groups, range of KPATH sequence numbers for the combining groups, the average percentage of overlap between the members of the combining groups, and the average percentage of overlap within the combined group.

INPUT/OUTPUT

Input a. KPATH file.

Output a. Printed group membership report.

b. Optionally, report placed on the REPORT file.

c. GRMBRS file for DIAGRM.

CONTROL

Section I: FILENAMES CARD

Columns	
1- 6	'GRMBRS'
8-18	Filename of the KPATH file.
20-30	Filename for the GRMBRS file.
32-42	Filename of the REPORT file.

Section II: CONTROL CARD

Columns	
1- 6	'GRMBRS'
8-15	Study ID.
17	Number of PAGE HEADING CARDS in Section III. 0-8 cards are allowed.
19-24	REPORT ID.
26-73	REPORT TITLE.
75	1 = Do not add report to the REPORT file. 0 = Add report to the REPORT file.
77	This option should not be used when Task Analysis data is being processed. When using the Profile Analysis system, this option may be necessary in order to reduce the size of 'Between' and 'Within' values for printing in the GRMBRS report and in the DIAGRM print which follows GRMBRS (values are reduced to a percentage of the largest Between value). The stage data output from the GROUP program should be referenced to determine if the 'Between' and 'Within' values can be printed. GRMBRS can print a value up to ± 99999.99 and DIAGRM can print a value up to ± 99.9 (actually ± 100.0 but it is printed as ± 99.9 to save a print position). If this option is necessary, punch '1' in column 77.

Section III: PAGE HEADING CARDS

Columns	
1-80	Any information desired.

GENERAL DESCRIPTION

The KPATH file is read and information stored in memory. Retained information is group stage number at which two groups combined, average-between, average-within, and KPATH sequence. The report is computed and printed from this information and added to the REPORT file if required. The GRMBRS file for DIAGRM is created as the report is printed.

MESSAGES

1. 'TAPE AND CNTRL CARD FAIL TO MATCH,

   TAPE = XXXXXXXX

   CARD = XXXXXXXX.'

   Self explanatory.

PROGRAM IDENTIFICATION

Name	GROUP
Language	FORTRAN V
Date Written	Aug 1973
Programmer	Barton

FUNCTION

This program uses the similarity matrix computed in the overlap (OVRLAP) program to form clusters of cases. The grouping technique, called 'collapsing the matrix' or 'hierarchical grouping,' involves repeated searching for those individuals or partially formed clusters which have the highest (or lowest) remaining similarity, depending on whether a 'maximizing' or 'minimizing' process was requested. The 'maximizing' option is always used for job survey data. Each new clustering or 'collapse' is called a 'stage' and the vectors of similarity values for the clusters being merged are combined according to a specified mathematical algorithm to form an integrated cluster. Several formulas for combining groups are available. The collapsing process continues until a single group has formed which contains all cases in a study.

INPUT/OUTPUT

Input	a. OVRLAP MATRIX file
Output	a. STAGE DATA file
	b. Optional printed Stage Data Summary

CONTROL

Section I FILENAMES CARD

Columns	
1- 5	GROUP
8-18	Filename of the OVRLAP MATRIX file
20-30	Filename for the STAGE DATA file

Section II CONTROL CARD

Columns	
1- 5	GROUP
8-15	Study I.D.
17	ƀ or 0 = Do not print summary
	1 = Print stage data summary
19	ƀ or 0 = Maximize
	1 = Minimize
21	1 = Collapse option 1
	ƀ or 2 = Collapse option 2

PROGRAM NOTES

GROUP starts with the NCASE rows of an OVRLAP output matrix. Each matrix row (or column) represents an incumbent; each row element is the value of the overlap between that row and column, i.e., between two incumbents. This time or task overlap is the 'similarity function,' or 'criterion value,' on which clustering (hierarchical grouping) of incumbents is decided. The extreme value of each row (maximum or minimum, depending on the option chosen) indicates to which other row this row is 'best' related. An extreme value and the corresponding row subscripts are entered for each row into a 'best table.' The extreme (under the same option) entry in the best table is called BEST; the numerically smaller of the two subscripts is IBEST, the other is JBEST. GROUP proceeds through a series of 'stages,' numbered from (NCASE-1) to 1. At each stage, the number of rows is reduced by one by an operation called 'collapse,' until at Stage 1 only a single row, the cluster of all incumbents, remains. Corresponding elements of the rows IBEST and JBEST are collapsed according to the following options:

Option 1 (when maximizing)

V'_{ik} = larger of V_{ik} and V_{jk}

Option 1 (when minimizing)

V'_{ik} = smaller of V_{ik} and V_{jk}

Option 2

$$V'_{ik} = \frac{V_{ik} W_i + V_{jk} W_j}{W_i + W_j}$$

where:

i = IBEST, the smaller row number

j = JBEST, the larger row number

k = the element number within a row; 1,2,...NCASE

V_{ik} = the old value from row i

V_{jk} = the old value from row j

V'_{ik} = the new value due to collapse of rows i and j (IBEST and JBEST). After collapse, row IBEST contains these new values, which are representative of both rows collapsed. Row JBEST is deleted; all references to the subscript JBEST are ignored, and the order of the matrix is thus reduced by one.

W_i = weight (number of incumbents) of row i before collapse.

W_j = weight of row j before collapse. After each collapse, W_i is the sum of the weights of both rows IBEST and JBEST. Initially, each weight is 1; at the final stage, W_1 equals NCASE.

At each stage, GROUP also calculates an 'average within' value for the cluster produced by row collapse. While BEST is the overlap between the two rows (clusters), IBEST and JBEST, before collapse, the average within is the average overlap among all incumbents within the new row IBEST after collapse, calculated by the following formula:

$$V'_{ii} = \frac{V_{ii}W_i^2 + V_{jj}W_j^2 + 2V_{ij}W_iW_j}{(W_i + W_j)^2}$$

where:

i = IBEST

j = JBEST

V_{ii} = previous average within for all rows collapsed into row i

V_{jj} = like V_{ii}, but for row j. Average within is initially 100% for each row.

V_{ij} = BEST

W_i = weight of row i before collapse

W_j = weight of row j before collapse

V'_{ii} = new average within for row i

GENERAL DESCRIPTION

During initialization, GROUP compares the Study I.D.'s from the control card and the OVRLAP matrix file, and aborts in case of no match. As the halfword-packed overlap rows are read from input onto random access disk, they also are unpacked and scanned for maximum (or, by option, minimum) values to be entered into the best table, one entry per row. For each stage, the BEST (criterion) value is extracted from the best table, and the associated rows, if necessary, are brought in from disk and updated. The rows, IBEST and JBEST, are collapsed into IBEST, all JBEST row and column elements are deleted, and stage data information is computed and written to the output file and optionally to the printer. The maximum (or minimum) value in IBEST replaces the old best table entry for that row (BEST); the JBEST row entry is deleted. For all other IBEST- or JBEST-subscripted best table entries, the corresponding rows are updated to the current stage, and the extreme row value is extracted and entered. Subsequently, similar steps are taken to insure that pairs of best table entries which are symmetric elements in the grouping matrix are equal and actually are the extreme values of their rows. If IBEST is not required for the next collapse it is written to disk, along with its current average within and a row update pointer. Processing continues until a single row remains; a message is printed to indicate successful program termination.

MESSAGES

1. 'OVERLAP TAPE STUDY I.D. DOES NOT MATCH CONTROL CARD STUDY I.D.
   OVERLAP TAPE STUDY I.D. IS XXXXXXXX AND CONTROL CARD STUDY I.D.
   IS XXXXXXXX. RUN ABORTS.' Self explanatory.
2. 'OVERLAP MATRIX ID IS XXXX
   XXXIMIZE
   GROUP OPTION = X
   STAGE IBEST JBEST MBRS I MBRS J BEST AVG'
   Optional stage data summary.
3. 'GROUP HAS COMPLETED XXXX STAGES FOR STUDY XXXXXX. GROUP ENDS.'
   Successful program completion.

PROGRAM IDENTIFICATION

Name	GRPDIF (Group Difference)
Language	FORTRAN V
Date Written	1966/May 1973
Programmer	CSC/Weissmuller

FUNCTION

This program calculates and reports the difference between two job descriptions in terms of percentage of members performing each task and/or average percent time spent. Difference values are presented in ascending or descending order on either value (from largest negative to largest positive difference or vice versa) or in task number order.

INPUT/OUTPUT

Input
- a. HISTORY/KPATH file.
- b. JOB DESCRIPTION file.
- c. REQUEST cards.

Output
- a. The difference report on the printer, and optionally added to the REPORT file.

CONTROL

Section I — FILENAMES CARD

Columns	
1- 6	GRPDIF
8-18	Filename of the HISTORY/KPATH file
20-30	Filename of the JOB DESCRIPTION file
32-42	Filename of the REPORT file

Section II IDENTIFICATION CARD

Columns	
1- 6	GRPDIF
8-15	Study ID
17	The number of PAGE HEADING cards that will be in Section III. 0-6 cards are allowed.

Section III PAGE HEADING CARDS

Columns	
1-80	Any information desired.

Section IV CONTROL CARD

Columns	
1- 5	This field will cause print suppression when the computed absolute difference in percent members performing, or the computed absolute difference in average percent time spent is less than the specified number. The sort sequence which is specified in column 9 will determine whether this field refers to average percent time spent or percent members performing. When this field is left blank or when sort option 3 is used, this field is set to .01 by the program. A decimal point must be used with any number punched in this field.
7	Print control option: 1 = Print only percent members performing data. When this option is used, column 9 must be 1, 3 or 4. 2 = Print only average percent time data. When this option is used, column 9 must be 2, 3 or 5. 3 = Print both percent members performing data and average percent time data. Column 9 may be 1 thru 5. (assumed option)

9 Sort Control.

1 = Sort in descending order on difference in percent of members performing. (assumed option)

2 = Sort in descending order on difference in average percent of time by all group members.

3 = No sort, output in task order.

4 = Sort in ascending order on difference in percent of members performing.

5 = Sort in ascending order on difference in average percent time by all group members.

11 Output Control.

Blank = Add this report to the REPORT file.

1 = Do not add this report to the REPORT file.

13-18 REPORT ID

Section V REQUEST CARDS (in pairs)

Columns

1-80 These are two system produced REQUEST cards defining the groups over which the difference will be taken.

The three cards defined by Sections IV and V constitute one control set for a single group difference. Any number of these three card sets can be stacked to produce multiple group difference reports.

GENERAL DESCRIPTION

The program processes control cards through Section V and obtains the first group difference set. The JOB DESCRIPTION file is searched for the two requested groups. These are brought into memory. The difference is calculated, sorted, matched with a task title and then output to the printer and the REPORT file if requested. The process continues until an EOF is encountered.

MESSAGES

1. 'TAPE-ID DOES NOT MATCH CONTROL CARD. TAPE ID IS XXXXXXXX.' Self explanatory.
2. 'NUMBER OF HEADING CARDS EXCEEDS 6.' 0-6 PAGE HEADING CARDS are allowed.
3. 'INVALID PRINT OPTION. ASSUMED 3.' Valid options are 1, 2, 3.
4. 'CANNOT FIND RECORD FOR REQUEST CARD Y IN THE ABOVE SET, REPORT SKIPPED.' Unable to locate matching record on the JOB DESCRIPTION file for this REQUEST card.
5. 'INVALID SORT OPTION. ASSUMED 1.' Valid options are 1, 2, 3, 4, 5.
6. 'INVALID PRINT/SORT OPTION COMBINATION. THIS REPORT IS BYPASSED.' Read the writeup. This error could be caused by the program taking an assumed SORT option.

PROGRAM IDENTIFICATION

Name	GRPSUM (Group Summary)
Language	FORTRAN V
Date Written	1966/May 1973
Programmer	CSC/Weissmuller

FUNCTION

Selects job descriptions from the JOB DESCRIPTION file and calculates either (1) the percent of members performing or (2) the average percent time spent by all members. The method is selected by control card. The summarized data will be printed in task order and the individual description reports are ordered according to the sequence of the input request cards. Task titles are printed with summary data. The report is produced on the printer and optionally on the REPORT file.

INPUT/OUTPUT

Input	a. HISTORY/KPATH file. b. JOB DESCRIPTION file. c. REQUEST cards.
Output	a. Summary report on the printer and, optionally, on the REPORT file.

CONTROL

Section I FILENAMES CARD

Columns	
1- 6	GRPSUM
8-18	Filename of the HISTORY/KPATH file.
20-30	Filename of the JOB DESCRIPTION file.
32-42	Filename of the REPORT file.

Section II	IDENTIFICATION CARD
Columns	
1- 6	GRPSUM
8-15	STUDY ID
Section III	CONTROL CARD
Columns	
1	1 = report 'duty' group summary only 2 = report 'task' group summary only (assumed option) 3 = report 'duty' and 'task' group summary
3	Type data to be reported 1 = percent of members performing (assumed option) 2 = average percent time (Only allows 6 request cards)
5	The number of PAGE HEADING CARDS that follow in Section IV. 0-6 cards are allowed.
7	Blank = Add this report to the REPORT file. 1 = Do not add this report to the REPORT file.
9-14	REPORT ID
16-63	REPORT TITLE
Section IV	PAGE HEADING CARDS
Columns	
1-80	Any information desired.
Section V	REQUEST CARDS
Columns	
1-80	A maximum of 12 REQUEST cards may be in one Section V set. If average percent time is reported, only 6 REQUEST cards are allowed per set.

Section VI END CARD

Columns

1- 3 END

Sections III through VI constitute one controlling set for one report. Any number of sets may be stacked to produce multiple reports.

GENERAL DESCRIPTION

The program reads the first control card and checks the HISTORY/KPATH file for correct study identification. The duty-task titles are read and saved for later use. The control cards for Sections II through VI are processed. The JOB DESCRIPTION file is searched for the records matching the REQUEST cards. Required data is accumulated and output to a work file. The detail report is produced from the work file data, formatted for printing, and output to the REPORT file if requested. When a report is completed, the program recycles to read another set of Section III-VI cards.

MESSAGES

1. 'ERROR IN CONTROL CARD/TAPE, TAPE LISTED BELOW XXXXXXX YYYYYYYY JOB ABORTED.' Tape and control card do not match. Where X is the program name and Y is the study identification from tape; the control card from the reader is printed immediately before error message.
2. 'OPTIONS IN CC 1, 3, 5, OR 7 NOT IN RANGE. REPORT SKIPPED.' Any invalid condition in a Section III card has caused that report to be skipped.
3. 'STUDY ID INCORRECT ON ABOVE CARD, CARD SKIPPED.' An incorrect REQUEST card has been placed in the input deck.

4. 'MORE THAN X REQUEST CARDS IN ABOVE SET, EXTRA CARDS SKIPPED.' See Section V.

5. 'CANNOT FIND REPORT WITH SEQUENCE XXXX THE ABOVE REPORT HAS BEEN SKIPPED.' The program was unable to locate some of the requested records on the JOB DESCRIPTION file.

PROGRAM IDENTIFICATION

Name	INPSTD (Input Standard)
Language	FORTRAN V
Date Written	1967/Apr 1973
Programmer	CSC/Stacey

FUNCTION

INPSTD creates the CODAP HISTORY tape. Input to this program is the SETCHKed raw data tape which contains response and history information. Input also includes Duty/Task title cards and History variable definitions. This program converts raw task responses to percentages, constructs data vectors for each case, reorganizes the data to the HISTORY tape format and writes the data to tape. INPSTD will accept 20000 cases, 1700 task variables per case, 1700 history variables per case, and 26 duty variables. INPSTD creates 9 computed variables and writes them to the HISTORY tape.

INPUT/OUTPUT

Input

a. SETCHKed raw data tape.

b. Filenames card and control card.

c. Task Response Weight and Substitution cards to adjust raw data responses (optional).

d. Format cards to describe the SETCHKed data.

e. Tasks within Duty cards.

f. Variable Definition cards.

g. Task/Duty Title cards.

Output

a. CODAP HISTORY tape.

CONTROL

Section I: FILENAMES CARD

Columns	
1- 6	"INPSTD"
8-18	Filename of the SETCHKed raw data tape.
20-30	Filename for the HISTORY tape.

Section II: CONTROL CARD

Columns	
1- 6	"SETSTD"
7-12	MAXTSK. Maximum acceptable value for a task response right adjusted. Maximum is 999999.
14-15	NCARD. Number of cards in a case right adjusted. Maximum is 50.
17-20	NTASK. Number of tasks in the inventory right adjusted. Maximum is 1700.
22-25	NHIST. Number of history variables in the inventory right adjusted. Maximum is 1700 (1700 computer words are reserved for history variables; therefore, 1700 variables are allowed only if all variables are 6 characters or less in size).
27-28	NDUTY. Number of duties in the inventory right adjusted. Maximum is 26.
30-37	SID. Study ID to be placed in the Communication Region of the HISTORY tape.
39-43	NCASE. The number of cases on the SETCHKed raw data tape. Maximum is 20000. (Only 7000 cases may be GROUPed.)

45 Weight or substitution option. Task responses may be adjusted by punching an option number 1, 2, 3, or 4. These options are discussed in Section III. If blank, Section III is skipped.

Section III: Weight or substitution cards (optional). These cards are used to adjust the raw data responses read from the SETCHKed tape. There are 4 options available. An option is selected by punching 1, 2, 3, or 4 into column 45 of the control card. The options are as follows:

1 = Ratio Interval Scale. One card is required. Columns 1 - 8 will contain an upper ratio value right adjusted. A decimal point may be used. The lower ratio value is set to 1 by the program. Example: For a 7-point scale the desired ratio of the highest possible rating (7) to the lowest (1) is set at 64 to 1 (64 being specified in columns 1 - 8). Program computation is: Base = $\sqrt[6]{64} = 2$. Substituted weights for scale values = 2^0, 2^1, 2^2, 2^3, 2^4, 2^5, 2^6 = 1, 2, 4, 8, 16, 32, 64.

2 = Exponential Scale. One card is required. Columns 1 - 8 will contain an exponent right adjusted. A decimal point may be used. For this option, each task response will be raised to the power specified in columns 1 - 8.

3 = Substitution Scale. In this option, weights are substituted for task responses. Each weight must be punched right adjusted in a 6 column field starting from column 1. A weight may contain a decimal point. There may be 13 weights punched per card and as many cards as required may be used. There must be as many 6 character fields as there are possible task responses. Six character fields correspond to task responses as follows: Field 1 (columns 1 - 6) would replace a "1" response, Field 2 (columns 7 - 12) would replace a "2" response, etc. A blank field replaces a task response with zero.

4 = Cross Products Scale. Weights are punched as in option 3. Each task response is multiplied by its corresponding weight and the product replaces the response.

NOTE: The maximum number of weights the program can handle is 200.

Section IV: FORMAT CARDS.

See 'FORMAT CARDS' in this manual.

Section V: Tasks within duty cards. These cards define the number of tasks within each duty. The cards must be in duty order such that the number N punched on the first card assigns task 1 through N to that duty; the number P punched on the second card assigns tasks (N + 1) through (N + P) to that duty and so on.

Columns	
4	The duty identification character.
5 - 8	The number of tasks in this duty, right adjusted.

Section VI: Variable definition cards. These are an ordered series of cards that form a variable dictionary. The program adds FORTRAN-like format characters at the end of the tape record when the titles are output. The formats are used in setting up control cards for various CODAP programs. Only one card per variable is permitted and since there is no other identification, the order of the cards must be sequential by history variable number. That is, card #1 is the title for history variable 001, card #2 is for history variable 002. Titles for computed variables are program generated and do not appear in these cards.

Columns	
1	V representing a history variable
2 - 4	Variable number
5	Blank
6	An "=" sign
7 - 80	The variable definition

Section VII: Task title cards. The data punched in Columns 7-72 will be printed on output listings verbatim.

Columns	
1 - 4	Not read by program. May contain any user data, usually sort control.
5	Card sequence number of a particular task title; there may be up to 9 title cards per task.
7	Duty character. Must be one of the characters given in Column 4 of Section V card.
8 - 11	Task number within duty, right adjusted with leading blanks.
13 - 72	Task title.

Section VIII: Duty title cards. These have the same layout as in Section VII except that Columns 8 - 11 must be blank.

ERAL DESCRIPTION

The control card is read. If the weighting option is requested, the weight cards are read and saved. The format cards are read and a FORTRAN format is generated and saved. The Communications Region is generated and written on the HISTORY tape. The Task within Duty cards are read and saved. The Relative Location Table and Task within Duty table are generated and written on the HISTORY tape. The Computed variable definitions and History variable definitions are written on the HISTORY tape. The Task and Duty titles are read, formatted, and saved for later use. Case data is read from the SETCHKed tape using the format generated by the format cards. Task response data is weighted (if requested) and converted to percentage values. The percentages are integer of the form XXX.XXX with the decimal implied. Computed variables, History variables, and percentage values are written on the HISTORY tape for each case. When all cases have been processed, the Task and Duty titles are written on the HISTORY tape. The Title locator table and Task within Duty table are generated and written on the HISTORY tape. An END OF DATA record is written on the HISTORY tape and the program terminates.

MESSAGES

1. "NTASK FROM THE CONTROL CARD IS XXXX. THE T COUNT FROM FORMAT CARDS IS XXXX. RUN ABORTED." Self explanatory.
2. "NHIST FROM CONTROL CARD IS XXXX. THE H COUNT FROM FORMAT CARDS IS XXXX. RUN ABORTED." Self explanatory.
3. "A CARD IS OUT OF SEQUENCE IN TASK TITLE NUMBER XXXX." Column 5 of the XXXXth task title read is out of sequence.
4. "FOR DUTY X, XXXX TASK TITLES WERE EXPECTED AND XXXX TITLES WERE PRESENT." The number of tasks read for Duty X does not agree with the count given in cols 5-8 of the tasks within duty card.
5. "ERROR IN DUTY/TASK TITLE CARDS. ABORTED." This message prints when all Duty/Task cards have been scanned, and error number 3 or 4 has occurred.
6. "DUTY/TASK TITLES EXPECTED XXXX. TITLES READ XXXX. RUN ABORTED." The sum of NTASK and NDUTY given on the control card does not agree with the number of Task and Duty titles read.

7. "CASE DATA OUT OF SEQUENCE. CASE XXXX FOLLOWED BY XXXX. RUN ABORTED." This message occurs only when format cards do not properly describe the SETCHKed raw data.
8. "CHECK CHARACTER NO MATCH. FORMAT CARD CHARACTER IS X. DATA CHARACTER IS X. RUN ABORTED." Check characters used in format cards as described in Section IV must match characters in corresponding columns of the SETCHKed raw data.
9. "THE WEIGHT AND SUBSTITUTION OPTION HAS RESET ALL RESPONSES FOR CASE XXXX TO ZERO. NOT ALLOWED. RUN ABORTED." Every data case must perform at least one task to remain in the system.
10. "CASES EXPECTED XXXX. CASES ENCOUNTERED XXXX. RUN ABORTED." The NCASE value from the control card does not agree with the number of cases encountered from the SETCHKed raw data tape.
11. "THE FORTRAN FORMAT GENERATED FROM FORMAT CARDS EXCEEDS ALLOCATED STORAGE. SIMPLIFY INPUT. RUN ABORTED." 1000 words are reserved to store the FORTRAN format. It has been exceeded.
12. "FORMAT CONTROL CARD DOES NOT BEGIN C... RUN ABORTED." Format cards are not in their proper position in the run stream, or they are incorrectly prepared.
13. "THE NUMBER OF FORMAT CARD CHECK CHARACTERS EXCEEDS 500. RUN ABORTED." Remove as many Format characters as required to satisfy the 500 character limit.
14. "A FORMAT CARD DOT (.) WAS USED INCORRECTLY. RUN ABORTED." Read Section IV of the write-up.
15. "MORE THAN ONE TASK LENGTH ON FORMAT CARDS. RUN ABORTED." The number of dots (.) following a "T" must remain constant for all tasks.

16. "THE NUMBER OF HISTORY VARIABLES DEFINED BY FORMAT CARDS EXCEEDS 1700. RUN ABORTED." 1700 History variables is maximum.
17. "THE NUMBER OF TASK VARIABLES DEFINED BY FORMAT CARDS EXCEEDS 1700. RUN ABORTED." 1700 Task variables is maximum.
18. "XXXX COMPUTER WORDS ARE REQUIRED TO STORE HISTORY VARIABLES. THIS EXCEEDS THE MAXIMUM OF 1700. RUN ABORTED." Reduce the number of History variables until the 1700 word restriction is satisfied.

PROGRAM IDENTIFICATION

Name	JDINDX (Job Description Index)
Language	FORTRAN V
Date Written	July 1973
Programmer	Weissmuller

FUNCTION

This program prints an index of the JOB DESCRIPTION file. Information printed is Study ID, NCASE, NDUTY, Filename, NTASK, Report ID, Source file, number of members, and Report number.

INPUT/OUTPUT

Input	a. JOB DESCRIPTION file
Output	a. Printed index

CONTROL

Section I FILENAMES CARD

Columns	
1- 6	JDINDX
8-18	Filename of the JOB DESCRIPTION file

GENERAL DESCRIPTION

The first JOB DESCRIPTION record is read and certain information common to all job descriptions is printed. The program then prints one line of information (as described above) for each job description and terminates. The Filenames card is the only control card required.

PROGRAM IDENTIFICATION

Name	JOBGRP (Composite Job Descriptions)
Language	FORTRAN V
Date Written	1966/Apr 1973
Programmer	CSC/Stacey

FUNCTION

This program calculates and prints composite job descriptions for groups formed during the hierarchical grouping process. Both duty and task job descriptions may be reported in high to low sequence of either "average percent time spent by all members" or "percent of members performing." A job description provides the following information: duty/task number, duty/task title, percent of members performing each duty/task, average percent time spent by members performing, average percent time spent by all members, and cumulative average percent time spent by all members. JOBGRP is limited to 7000 cases since that is the maximum allowed by OVRLAP/GROUP.

INPUT/OUTPUT

Input	a. KPATH file.
	b. Control card(s).
Output	a. JOB DESCRIPTION file.
	b. REPORT file.
	c. Printed report(s).
	d. REQUEST card(s).

CONTROL

Section I: FILENAMES CARD

Columns	
1- 6	"JOBGRP"
8-18	Filename of the KPATH file
20-30	Filename of the JOB DESCRIPTION file
32-42	Filename of the REPORT file

Section II: IDENTIFICATION CARD

Columns	
1- 6	"JOBGRP"
8-15	Study ID
17	Number of PAGE HEADING cards following. 0-3 PAGE HEADING cards are allowed.

Section III: PAGE HEADING CARD(S)

Columns	
1-80	Any information the user desires. PAGE HEADING information is printed on the first page of a job description.

Section IV: CONTROL CARD(S)

Columns	
1- 4	LOW group (stage) to be included in this job description.
6- 9	HIGH group (stage) to be included in this job description. The LOW and HIGH fields form an inclusive range to allow processing of multiple job descriptions. When the HIGH field is blank, LOW is assumed.

11	'1' = Print only DUTY descriptions '2' = Print only TASK descriptions '3' = Print both DUTY and TASK descriptions. When this field is blank, '3' is assumed. Both DUTY and TASK data is added to the JOB DESCRIPTION file even when option '1' or '2' is requested.
13	'1' = Do not add these reports to the REPORT file or to the JOB DESCRIPTION file. 'blank' = Add these reports to the REPORT file and to the JOB DESCRIPTION file.
15-18	Minimum number of cases to be included in these job descriptions. Assumes 5 if blank. Job descriptions are skipped when Minimum is not satisfied.
20-23	Maximum number of cases to be included in these job descriptions. Assumes NCASE if blank. Job descriptions are skipped when Maximum is exceeded.
25-28	Maximum number of tasks to print. Assumes NTASK if blank. The REPORT file and the JOB DESCRIPTION file always contain complete descriptions even when 'Maximum' is given.
30-32	Maximum 'Cumulative Percentage' to print. Assumes 100 if blank. The REPORT file and the JOB DESCRIPTION file always contain complete descriptions even when 'Maximum' is given.

34 '1' = Sort from high to low on 'Percent of Members Performing'.

blank = Sort from high to low on 'Average Percent Time Spent By All Members'.

NOTE: The Section IV (Control Card) may be repeated any number of times.

GENERAL DESCRIPTION

Information from the KPATH file is loaded on disk. Disk is searched and job descriptions are computed, printed, and added to the REPORT and JOB DESCRIPTION file for each group (stage) defined by the control card. When the HIGH range has been satisfied, the program reads another control card and recycles. If there are no more control cards the program terminates.

MESSAGES

1. 'STUDY ID NO MATCH. TAPE IS XXXXXXXX. CARD IS XXXXXXXX.' The Study ID from the KPATH file does not match the Study ID given on the control card.
2. 'GROUP XXXX HAS MBRS = XXXX. MIN = XXXX, MAX = XXXX.' The job description for this group was not reported because the Minimum/Maximum limits as specified on the control card were not satisfied.

PROGRAM IDENTIFICATION

Name	JOBIND
Language	FORTRAN V
Date Written	Nov-Dec 1973
Programmer	Barton

FUNCTION

From groups represented by request cards, the program prints a job description with specified background information for each individual in the group. The size and sequence of these individual reports can be specified.

INPUT/OUTPUT

Input
a. History/KPATH file
b. JOB DESCRIPTION file

Output
a. Individual job descriptions on the printer, and optionally added to the REPORT file with punched request cards.

CONTROL

Section I: FILENAMES CARD

Columns	
1-6	"JOBIND"
8-18	Filename of the History/KPATH file
20-30	Filename of the JOB DESCRIPTION file
32-42	Filename of the REPORT file (optional)
44-54	Filename for the JOBIND TRACE file (optional)

Section II: IDENTIFICATION CARD

Columns	
1-6	"JOBIND"
8-15	Study I.D.

Section III: SET CARD (Print, Sort, and Report options)

Columns	
1-3	"SET"
5	Number of heading cards following: blank, 1, 2, or 3
7	'b̸' = Print full report '1' = End report after printing the number of tasks specified in cc 9-12 '2' = End report after printing the number of pages specified in cc 9-12 '3' = End report after printing the task that causes the "cumulative per cent" to exceed the percentage specified in cc 9-12 (integer per cent, 0 to 100)
9-12	Termination value, right justified (see cc 7)
14	'b̸' = Print Duty - Task I.D. '1' = Print sequential task numbers
16	Number of copies: min. (default) = 1, max. = 9
18	'b̸' = Print in same order as input sequence '1' = Print in random sequence '2' = Print in ascending sequence sorted on major (cc 20-23) and minor (cc 25-28) sort fields

20-23	Variable I.D. for major sort field (cc 18=2); blank for no sort
25-28	Variable I.D. for minor sort field (cc 18=2); blank if no minor field
30	'b̸' = Write to REPORT file '1' = Do not use REPORT file
32-37	Report I.D. for all reports produced by this set
39-80	Report title
Section IV:	HEADING CARD(S) (0 to 3)
Columns	
1-80	Any desired text excluding '@' in cc 1
Section V:	VAR CARD (Optional)
Columns	
1- 3	"VAR"
5- 6	Number of SPEC cards following (max. = 30)
8-11	Variable I.D. (Vxxx) for visual control field on JOBIND punched (output) request card
13	'1' = Print "JOB NUMBER" after the heading cards of each report. The JOB NUMBER is original sequence of the case within the current group being printed. The number goes from 0001 to XXXX where XXXX is the number of members being printed. This number is printed in a box of asterisks. 'b̸' = Do not print JOB NUMBER

15-18	Job number bias. The number in these columns will be added to the JOB NUMBER before printing. If blank, JOB NUMBER will go from 0001 to XXXX as above. If, for example cc 15-18 contained '0005', JOB NUMBERS would go from 0006 to YYYY where YYYY = XXXX + 5.
20	'b̸' = Do not use any random sampling '1' = Take a random sample of the specified group. Select the percentage of the group specified in cc 22-24. '2' = Take a random sample of the specified group. Select the number of cases specified in cc 22-24.
22-24	Random Sample Number. If cc 20 = '1': Specify a percent from 001 to 100 If cc 20 = '2': Specify number of members to select from 001 to 999
26	'1' = Change JOB NUMBER to CASE CONTROL number V001. Ignore cc 15-18 'b̸' = Use normal JOB NUMBER option
28	'1' = Record this sort on the TRACE file. 'b̸' = Do not record this sort on the TRACE file.
Section VI	SPEC CARD(S) (0 to 30)
Columns	
1- 4	Variable I.D. (Vxxx or Cxxx)
6- 7	Number of DECODE cards following (max. = 99) Only background (Vxxx) variables may be decoded, and only the first 6 characters may be decoded into 18 characters.

9-26	Variable title to be written with the value/decode.

NOTE: If any SPEC cards are used, they must follow the VAR card.

Section VII: DECODE CARD(S) (0 to 99)

Columns	
1-6	Value of variable exactly as it appears on the tape
8-25	Title to replace the above value

NOTE: If any DECODE cards are used, they must follow the appropriate SPEC card.

Section VIII: CODAP REQUEST CARD (Reference Layout only)

Columns	
1-4	Study from the Study I.D. of the file
5-10	Report I.D.
11-58	Report Title
77-80	Report ('J-sequence') number

NOTE: The card is punched by JOBSPC or JOBGRP; request cards punched by other programs are NOT acceptable as they do not represent groups.

Section IX: END CARD

Columns	
1-3	"END"

NOTE: There must be one END card for each SET card. Runs may be stacked by repeating Sections III through IX.

GENERAL DESCRIPTION

On the first pass, JOBIND assigns files, verifies Study I.D., and extracts locator tables and task titles from the History/KPATH file. On each pass (control cards SET through END), the program checks for valid control fields and sets flags. The requested group is selected with Subroutines GETMSK and USEMSK, and the needed variable values and/or decodes, task responses, and their titles are written to a scratch file and optionally to the REPORT file, case by case. The order of these individual job descriptions may be the same as that on the History/KPATH file, or the reports may be sorted, either in ascending sequence on selected variables, or randomly. From the scratch file, multiple printed copies of the report set can be made. The program then seeks another SET card to begin the next pass; at end of file, JOBIND issues a closing message and stops.

MESSAGES

1. 'STUDY I.D. NO MATCH BETWEEN CONTROL CARD (XX-XX) AND COMMUNICATION REGION (XX-XX). JOBIND ABORTS.' Self-explanatory.
2. '"XXX" CARD IGNORED. SEARCHING FOR "END" CARD' Enough control cards have been encountered; extra cards are ignored.
3. 'CONTROL CARD ERROR ON "XXX" CARD. SEARCHING FOR "END" CARD.' This report set will be omitted due to control card error.
4. 'XXX IS INVALID VARIABLE SPECIFICATION. SKIPPING TO NEXT REPORT.' This report (set) will be omitted due to bad variable name given.
5. '"END OF DATA" NOT FOUND AFTER XX-XX CASES. JOBIND ABORTS.' Inaccurate NCASE on History/KPATH file.
6. 'FORMAT ERROR IN DICTIONARY ENTRY: XX - - - XX JOBIND ABORTS.' Bad dictionary record on History/KPATH file.
7. 'JOBIND COMPLETE FOR STUDY XX-XX. JOBIND ENDS.' The program has terminated successfully.

PROGRAM IDENTIFICATION

Name	JOBINV (Job Inventory)
Language	FORTRAN V
Date Written	1969/May 1973
Programmer	CSC/Stacey

FUNCTION

To generate a report containing a listing of duties and task titles included in a job inventory.

INPUT/OUTPUT

Input	a. HISTORY/KPATH file.
	b. A control deck.
Output	a. Printed report.
	b. Report added to the REPORT file.

CONTROL

Section I: FILENAMES CARD

Columns	
1- 6	'JOBINV'
8-18	Filename of the HISTORY/KPATH file.
20-30	Filename of the REPORT file.

Section II: CONTROL CARD

Columns	
1- 6	'JOBINV'
8-15	Study Identification, eight characters (alphanumeric).
17	Number of PAGE HEADING CARDS which will be in Section III. (0-8 cards may be used)
19	'1' = do not add report to the REPORT file. '0' = add report to the REPORT file.

Section III: PAGE HEADING CARDS

Columns	
1-80	Any information desired. 0-8 cards may be used.

GENERAL DESCRIPTION

The program reads the control card and compares the study ID obtained from the input file with that on the control card. If correct, the page heading cards are read and printed. The Title-Locator Table and the Tasks-Within-Duty Table are read from the input file. Using these tables, the Duty and Task Titles are brought into memory, formatted, and output in report form to the required devices. If the report was added to the REPORT file, a REQUEST card is punched.

MESSAGES

'CONTROL CARD ERROR. STUDY IDENTIFICATION 'XXXXXXXX' SHOULD BE 'YYYYYYYY' AS ON HISTORY DATA TAPE.' Correct the study ID on the control card or mount the input tape with the proper study ID.

PROGRAM IDENTIFICATION

Name	JOBSPC (Composite Job Descriptions)
Language	FORTRAN V
Date Written	1966/May 1973
Programmer	CSC/Weissmuller

FUNCTION

This program calculates and prints composite job descriptions for special groups whose membership is defined in terms of some combination of background or computed variables. Both duty and task job descriptions may be reported in high to low sequence of either average percent time spent by all members or percent of members performing. (A duty is a functional area comprising a number of tasks and, hence, is a summary report.) A job description provides the following information: duty/task number, duty/task title, percent of members performing each duty/task, average percent time spent by members performing, average percent time spent by all members, and cumulative average percent time spent by all members.

INPUT/OUTPUT

Input	a. HISTORY/KPATH file.
Output	a. JOB DESCRIPTION file.
	b. REPORT file.
	c. Printed reports.

CONTROL

Section I FILENAMES CARD

Columns	
1- 6	'JOBSPC'
8-18	Filename of the HISTORY/KPATH file.

20-30	Filename of the JOB DESCRIPTION file.
32-42	Filename of the REPORT file.
Section II	IDENTIFICATION CARD
Columns	
1- 6	'JOBSPC'
8-15	Study ID
17	Number of PAGE HEADING cards in Section III. 0-3 cards are allowed.
Section III	PAGE HEADING CARDS
Columns	
1-80	Any information desired.
Section IV	REPORT ID AND REPORT TITLE CARD
Columns	
1- 6	REPORT ID
8-55	REPORT TITLE
57	Selection type. Blank = Use Section VA for regular JOBSPC sample selection. '1' = Use Section VB for individual sample selection.
Section V	SAMPLE SELECTION
VA:	Regular Sample Selection. See 'SAMPLE SELECTION' in this manual
	OR
VB:	Individual Sample Selection

NOTE: This section (VB) may be used only if CC 57 above = '1'.

Part 1 VARIABLE ID CARD (1 required)

Columns	
4- 7	Variable name. Only background variables (VXXX) may be used as this method checks for EXACT MATCHES.
9	Blank = Variable format is A6 or less. '2' = Variable format is greater than 1A6. This will cause an exact match test on 12 characters.

Part 2 EXACT MATCH CARDS (1-1000 cards)

Columns	
1-12	Exact value of the variable above as it will appear on tape. If CC 9 is blank, only the first 6 columns will be used. Due to the variable format, only one value is allowed per card. A maximum of 1000 cards is allowed.

Part 3 END CARD

Columns	
1- 6	'END ' This end card tells the program that all the EXACT MATCH CARDS have been read.

Section VI CONTROL CARD

Columns	
1	1 = PRINT duty descriptions only. 2 = PRINT task description only. 3 = PRINT both task and duty description. When this field is blank, 3 is assumed. Both duty and task data is added to the JOB DESCRIPTION file even when option 1 or 2 is requested.

3- 7	Minimum number of cases permitted in this job description. (Description skipped if minimum not reached.) Assumes 5 if blank for regular specials, 1 for individual selections.
9-13	Maximum number of cases permitted in a job description. (Description skipped if maximum exceeded.) Assumes NCASE if blank.
15-18	Maximum number of tasks to print before terminating output on the printer. The REPORT file and JOB DESCRIPTION file always contain complete descriptions. Assumes NTASK if blank.
20-22	Maximum cumulative percentage to print before terminating output on the printer. The REPORT file and JOB DESCRIPTION file always contain complete descriptions. Assumes 100 if blank.
24	Sort Control 0 = Sort from high to low on Average Percent Time Spent by all Members 1 = Sort from high to low on Percent of Members performing.
26	1 = Do not add the report to the REPORT or JOB DESCRIPTION file. 0 = Add the complete job description to both files.

NOTE: Sections IV-VI may be repeated for multi-report runs.

GENERAL DESCRIPTION

The HISTORY/KPATH file is read and task data for selected cases is added to the current description being calculated. JOBSPC then reads the next set of oontrol cards and computes the next job description. As many job descriptions as storage allows are computed in one pass of the HISTORY/KPATH file. When the maximum number of descriptions have been formed, or if there is no text set of special job description cards, the current set of descriptions is printed.

MESSAGES

1. 'TAPE-ID MISMATCH. TAPE = XXXXXXXX. CARD = XXXXXXXX.' Self explanatory.
2. 'INVALID SORT OPTION IN CC 24. SET = 0 FOR XXXXXX.' CC 24 of the Section VI card has been set to 0.
3. 'INVALID BINARY/BCD TAPE OPTIONS - DESCRIPTION SKIPPED.' Valid option for Section VI, CC 26 is 0 or 1.
4. 'INVALID DESCRIPTION OPTION IN CC 1. SET = 3.' CC 1 of Section VI card must be 1, 2, or 3.

PROGRAM IDENTIFICATION

Name	KPATH (Create KPATH file)
Language	FORTRAN V
Date Written	1966/May 1973
Programmer	CSC/Weissmuller

FUNCTION

Creates the KPATH file from the HISTORY file using a sequence computed from the Stage Data. Cases on the KPATH file are arranged in such a way that each pair of individuals or groups which were merged during the grouping process will have a contiguous block of KPATH sequence numbers.

INPUT/OUTPUT

Input	a. HISTORY file.
	b. STAGE DATA file.
Output	a. KPATH file.

CONTROL

Section I: FILENAMES CARD

Columns	
1- 5	'KPATH'
8-18	Filename of the HISTORY file.
20-30	Filename of the STAGE DATA file.
32-42	Filename for the KPATH file.

Section II: IDENTIFICATION CARD

Columns	
1- 5	KPATH
8-15	Study ID.

GENERAL DESCRIPTION

KPATH reads the STAGE DATA file, constructs the KPATH sequence and stores BEST and AVERAGE WITHIN on a work file. It copies invariant information from the HISTORY file to the KPATH file and stores the case data records from the HISTORY file on disk in KPATH sequence. The data from disk is then copied to the KPATH file in KPATH sequence, with BEST and AVERAGE WITHIN values inserted. Finally, the duty/task titles are transcribed. The KPATH file contains all the original data from the HISTORY file, the case data records are now in KPATH order, and specific additional control values have been added.

MESSAGES

1. 'ID. ERROR, CRD = NNNNN AAAAAAAA
   TP = XXXXX. RUN TERMINATED.' Study ID from the HISTORY file does not match card.
2. 'STAGE DATA TAPE-ID = XXXXXXXX. HISTORY TAPE ID = XXXXXXXX.' Study ID's do not match. Wrong tapes being used.
3. 'KPATH SEQUENCE, TAPE SEQUENCE ERROR, RUN ABORTED.' Data error reading records from disk. Try again.
4. 'COMPARE ERROR STG CARD = AAAAAAA MMMMMM XXXX CORE = AAAAAAAA MMMMMM XXXX.' Error comparing STAGE DATA file to HISTORY file. Study identification error or STAGE DATA out of sequence. Job aborted.
5. 'NUMBER OF RECORDS ON TAPE AND STAGE CARDS DIFFER, RUN ABORTED.' Incorrect STAGE DATA file has been used.

PROGRAM IDENTIFICATION

Name	MTXPRT (Print overlapped matrix)
Language	FORTRAN V
Date Written	1966/May 1973
Programmer	CSC/Stacey

FUNCTION

This program calls for the subroutine OVRLAP to overlap all possible pairings of a set of composite job descriptions and then uses the program MTXPRT to print the between-group overlap values in matrix form. Overlap may be computed in terms of average percent time spent on tasks or in terms of number of tasks performed in common. The maximum number of groups that can be input to MTXPRT is 100.

INPUT/OUTPUT

Input	a. JOB DESCRIPTION file. b. REQUEST cards.
Output	a. Printed report. b. Optional copy of report on the REPORT file.

CONTROL

Section I: FILENAMES CARD

Columns	
1- 6	'MTXPRT'
8-18	Filename of the JOB DESCRIPTION file.
20-30	Filename of the REPORT file.

Section II: CONTROL CARD

Columns	
1- 6	'MTXPRT'
8	Number of PAGE HEADING CARDS in Section III. 0-6 cards are allowed.
10	1 = Compute a task matrix. 0 = Compute a time matrix.
12	1 = Do not add report to the REPORT file. 0 = Add report to the REPORT file.
14	1 = FORTRAN 'E' print format desired ± 0.XXXXXE±YY 0 = FORTRAN 'F' print format desired ±XXXX.XXXX
16-21	REPORT ID.
23-70	REPORT TITLE.

Section III: PAGE HEADING CARDS

Columns	
1-80	Free form description, any alphanumeric headings.

Section IV: REQUEST CARDS

Columns	
1-80	Contains REQUEST cards for selecting the records from the JOB DESCRIPTION file used to build the input matrix for overlapping. REQUEST cards are input in random order. A maximum of 100 cards is permitted. A major reduction in run time can be effected if the REQUEST cards are sorted on columns 78, 79, 80 into ascending order.

GENERAL DESCRIPTION

Data is retrieved from a JOB DESCRIPTION file through the use of REQUEST cards and placed on disk in a form to be overlapped. The overlap subroutine then overlaps the data in the normal CODAP manner, places the overlapped matrix on disk, then prints the overlapped matrix in report form.

MESSAGES

1. 'NUMBER OF REQUEST CARDS EXCEEDS MAX. 100 CASES WILL BE PROCESSED.' Number of REQUEST cards is greater than 100. The first 100 request cards will be processed.
2. 'TAPE SEQUENCE NO. AND CARD SEQUENCE NUMBER DO NOT MATCH. THIS REQUEST OMITTED. CSEQ=XXXX, SEQ=XXXX.' Cannot locate the sequence number given on a REQUEST card.
3. 'TAPE REPORT ID AND CARD REPORT ID DO NOT MATCH. THIS REQUEST OMITTED. CREP=XXXXXX, TREP=XXXXXX.' REPORT ID from the tape record sequence number requested does not match that on the REQUEST card. Probably wrong REQUEST card being used.
4. 'ERROR ROW ID OUT OF SEQUENCE.' 'I' row in overlap routine is wrong. Probably disk is not properly loaded.
5. 'J ROW ID OUT OF SEQUENCE.' 'J' row in overlap routine is wrong. Probably disk is not properly loaded.

PROGRAM IDENTIFICATION

Name	OVRLAP (Overlap)
Language	FORTRAN V
Date Written	Mar 1973
Programmer	Barton

FUNCTION

This program performs either a task or time overlap of task data from a HISTORY file and writes the overlap matrix to another file for program GROUP.

INPUT/OUTPUT

Input	a. HISTORY file.
Output	a. OVRLAP MATRIX file.

CONTROL

Section I FILENAMES CARD

Columns	
1- 6	OVRLAP
8-18	Filename of the HISTORY file.
20-30	Filename for the OVRLAP MATRIX file.

Section II CONTROL CARD

Columns	
1- 6	OVRLAP
8-15	Study I.D.
17	Blank = Compute a TIME matrix. 1 = Compute a TASK matrix.

PROGRAM NOTES

A blank in cc 17 of the Control Card causes a TIME matrix to be produced; each element is generated by the following formula:

$$A_{ij} = \sum_{k=1}^{n} \min [P_{ik} : P_{jk}] , i \neq j$$

where:

A_{ij} = the $(i,j)^{th}$ matrix element (i.e., the i^{th} row, j^{th} column)

P_{ik} = the k^{th} data element of vector (case) i

P_{jk} = the k^{th} data element of vector j

n = the number of possible responses per input case

A 1 in cc 17 causes a TASK matrix to be produced as follows:

$$B_{ij} = \frac{1}{2} \cdot \left[\frac{C_{ij}}{C_i} + \frac{C_{ij}}{C_j} \right] . 100, \quad i \neq j$$

where:

B_{ij} = the $(i,j)^{th}$ matrix element

C_{ij} = the number of common non-zero tasks between cases (data vectors) i and j

C_i = the number of non-zero tasks in vector i

C_j = the number of non-zero tasks in vector j

Note that for either option the matrix produced is symmetrical.

GENERAL DESCRIPTION

OVRLAP first reads the Study I.D. and Matrix Option from the control card, and sets flags for creating the chosen type of matrix. Then the communications region of the HISTORY file is read in, Study I.D. is compared with control card Study I.D., and the program aborts if the two do not match. With the addition of the matrix type (TIME or

TASK), the communications region is written to the output file; a unit weight vector is also generated and written to output. Next, task data are read from the HISTORY file, but unless the number of cases agrees with information from the communications region the run aborts. The program generates an overlap matrix row by row, writing each row as it is generated to the output tape in a halfword format. A closeout message is printed when the run is completed.

MESSAGES

1. 'HISTORY TAPE STUDY I.D. DOES NOT MATCH CONTROL CARD STUDY I.D.. HISTORY TAPE STUDY I.D. IS XXXXXXXX AND CONTROL CARD STUDY I.D. IS XXXXXXXX. RUN ABORTS.' Self-explanatory.
2. 'END-OF-DATA DISCREPANCY FOR NCASE = XXXX. RUN ABORTS.' After reading NCASE (number of cases, from communications region of HISTORY file) records, the next record was not END OF DATA.
3. 'OVERLAP COMPLETE FOR STUDY XXXXXXXX. OUTPUT FILE CONTAINS A XXXX BY XXXX XXXXXX MATRIX. OVRLAP ENDS.' Successful termination of run.

PROGRAM IDENTIFICATION

Name	PRDICT (Print Variable Dictionary)
Language	FORTRAN V
Date Written	1966/Apr 1973
Programmer	CSC/Stacey

FUNCTION

To print a report containing the descriptive titles and formats of all computed and history variables for a given survey. The formats obtained are used in preparing control information for various programs in the CODAP system.

INPUT/OUTPUT

Input	The HISTORY file created by program INPSTD or the KPATH file created by program KPATH.
Output	A dictionary report on the printer and, optionally, on the REPORT file. A request card is produced for each report added to this file.

CONTROL

Section I: FILENAMES CARD

Columns	
1- 6	'PRDICT'
8-18	Filename of the HISTORY or KPATH file.
20-30	Filename of the REPORT file.

Section II: CONTROL CARD

Columns	
1- 6	PRDICT
8-15	Study identification, eight characters (alphanumeric).
17	The number of PAGE HEADING cards that follow. 0-8 PAGE HEADING cards are allowed.
19	1 = do not add the report to the REPORT file. 0 = add the report to the REPORT file.

Section III: PAGE HEADING CARDS

Columns	
1-80	These cards are the page heading information. The headings, one line per card, may be any English description desired and will be printed sequentially on the report, single spaced. A maximum of eight heading lines is permitted.

GENERAL DESCRIPTION

The program reads the control card and compares the Study ID obtained from the input file with that on the control card. If correct, skips forward on the input file to find the dictionary records. These are brought into memory, formatted, and output in report form to the printer and REPORT file if requested.

MESSAGES

1. 'STUDY ID NO MATCH. RUN ABORTED.' The Study ID of the input file does not match the Study ID given on the control card.
2. 'MORE THAN 8 PAGE HEADING CARDS SPECIFIED. RUN ABORTED.' Self explanatory.

PROGRAM IDENTIFICATION

Name	PRIJOB (Primary Job Type Identifiers)
Language	FORTRAN
Date Written	1969/May 1973
Programmer	CSC/Weissmuller

FUNCTION

This program calculates and prints a report of those tasks which are determined to be "primary identifiers" of job types. Primary identifiers may be defined as the top x-number of tasks in a group job description in terms of percent of members performing or average percent time spent by all members. Primary identifiers may also be defined as those tasks performed by a specified minimum percentage of the group members or which exceed a specified average percent time spent value. This program allows a number of groups to be aligned in a single report for comparative purposes.

INPUT/OUTPUT

Input
- a. HISTORY/KPATH file
- b. JOB DESCRIPTION file

Output
- a. Printed output with report optionally added to the REPORT file

CONTROL

Section I FILENAMES CARD

Columns	
1-6	'PRIJOB'
8-18	Filename of HISTORY/KPATH file
20-30	Filename of the JOB DESCRIPTION file
32-42	Filename of the REPORT file

CONTROL (continued)

Section II: IDENTIFICATION CARD

Columns	
1-6	'PRIJOB'
8-15	Study Identification, eight characters (alphanumeric).
17	1 = Print task titles on 1st page of each report only. 0 = Print task titles on each page of each report.

Section III: OPTIONS CARD

Columns	
1-6	REPORT ID
8-55	REPORT TITLE
57	Number of PAGE HEADING cards in Section IV. Must be 0, 1, 2, or 3 (numeric).
59	0 = Add the report to the REPORT file. 1 = Do not add the report to the REPORT file.
61	Method of selecting primary identifiers. 1 = Percent of members performing. 2 = Average percent time spent by all members.
63-65	Number of primary identifiers to select from job description to comprise this report. Maximum of 465. If Columns 63-65 are non-zero, Columns 67-71 must be blank. Program will select the top 'K' tasks as primary identifiers, where 'K' is the value in Columns 63-65.

CONTROL (continued)

Section III: OPTIONS CARD (continued)

Columns	
67-71	The minimum task value to begin selecting primary tasks. If Columns 67-71 are non-zero, Columns 63-65 must be zero or blank. Program will select all tasks with task values ≥ X as primary identifiers where 'X' is the value in Columns 67-71. May be decimal number, or integer right adjusted in Columns 67-71.

Section IV: PAGE HEADING CARDS

Columns	
1-80	Page heading information. These are free form and may contain any alphanumeric description.

Section V REQUEST CARDS

Columns	
1-80	These are the system produced REQUEST CARDS. A maximum of 100 request cards may be in one Section V set.

Section VI: REPORT SEPARATION CARD

Columns	
1-3	'END'

Sections III through VI constitute one controlling set for one report. Any number of sets may be stacked to produce multiple reports.

GENERAL DESCRIPTION

The program reads the first control card and checks the HISTORY/KPATH file for correct study identification. The OPTIONS card is read and checked for input errors, then the PAGE HEADING cards are read. The task/duty titles are read from the HISTORY/KPATH file and stored on disk and the Title-Locator Table is read into memory. The set of REQUEST cards is read and the JOB DESCRIPTION file is searched for the records matching the REQUEST cards. Primary identifiers may be selected by either (1) specifying the top 'K' number of primary tasks desired, or (2) specifying the minimum task value to use in selecting primary tasks. By method (1), as each job description is found, the primary identifiers are selected, beginning with the largest task value in terms of either percent of members performing or average percent time spent by all members. The remaining task values are searched repetitively, each pass selecting the next largest value, until the desired number of identifiers are found. By method (2), all of those tasks whose task values are greater than or equal to the input value, are selected as primary identifiers.

An array of task numbers defining these primary identifiers is constructed, eliminating duplicate task numbers, and the necessary information for each job description is stored on disk in the order in which they are requested. When all of the request cards have been processed, the complete array of task numbers which define the primary identifiers are sorted into task order. The reports are formatted and output to the printer and, optionally, to the REPORT file. Each time an END card is read, the necessary end-of-block information is written on the REPORT file, a REQUEST card is punched for extracting the report by other programs, and the program recycles to read another set of Section III through VI cards.

MESSAGES

1. ERROR IN CONTROL CARD. TAPE IS PRIJOB XXXXXXXX JOB ABORTED. Study ID from control card does not match HISTORY/KPATH file.
2. INCORRECT VALUE FOR XXXX. MUST BE I, J, K. JOB ABORT. The indicated option is outside the allowable range, where XXXX is the option in error, and I, J, K are the values allowed as input. Correct the specified option on the Section III OPTIONS card.
3. STUDY ID INCORRECT. XXXX SHOULD BE YYYY. REPORT SKIPPED. The Study ID from the previous REQUEST card does not match the Study ID from the control card and the HISTORY/KPATH file.
4. CANNOT FIND REPORT XXXXXX, TAPE SEQUENCE NO. JJJJ. THIS REPORT SKIPPED. The report specified on a REQUEST card was not found on the JOB DESCRIPTION file. XXXXXX is the report IDSP and JJJJ is the sequence number punched on the REQUEST card.
5. NUMBER OF PRIMARY IDENTIFIERS FOR REPORT XXXXXX, SEQUENCE NO. JJJJ EXCEEDS 465. THIS REPORT SKIPPED. Dimension limitations do not permit descriptions of this size. Minimum task value may be increased to decrease the number of primary identifiers.
6. NO TASK VALUES FOUND IN REPORT XXXXXX - SEQUENCE JJJJ WHICH SATISFY THE INPUT MINIMUM TASK VALUE XXX.X. PROCESSING WILL NONE-THE-LESS BE ATTEMPTED. All task values for this job description were less than the input minimum task value. No primary identifiers were found.

MESSAGES (continued)

7. EITHER COLUMNS 63-65 OR COLUMNS 67-71 MUST BE NON-ZERO BUT NOT BOTH. JOB ABORT. Check instructions for Section III OPTIONS card

8. ALL OF THE XXX REPORTS REQUESTED FAILED TO HAVE ANY TASKS WHICH SATISFIED THE INPUT MINIMUM TASK VALUE. No reports generated.

PROGRAM IDENTIFICATION

Name	PROGEN (Program Generator)
Language	FORTRAN V
Date Written	1969/Aug 1973
Programmer	Adams/Weissmuller

FUNCTION

Generates coding from user supplied Verbs and FORTRAN statements to perform various operations on the HISTORY/KPATH file. Its primary use is to add new computed and/or History variables to the HISTORY/KPATH file.

INPUT/OUTPUT

Input

a. HISTORY/KPATH file.
b. FORTRAN statements, Verbs, and other control information from cards.
c. Any other input required by the generated program.

Output

a. New HISTORY/KPATH file (optional).
b. Variable Dictionary printout.
c. Generated program element and compiler listing (optional).
d. Any other output provided by the generated program.

CONTROL

Section 0 PROCESSOR CARD

Columns	
1- 9	@Z.PROGEN
11-17	,PROGEN

Section I FILENAMES CARD

Columns	
1- 6	PROGEN
8-18	Filename of the input HISTORY/KPATH file.

Section II	CONTROL CARD
Columns	
1- 6	PROGEN
8-15	Study ID
17-18	Number of computed variables (Cxxx) being added to the HISTORY/KPATH file. This number indicates the number of computed variable definition cards in Section III.
20-21	Number of History Variables (Vxxx) being added to the HISTORY/KPATH file. This number indicates the number of pairs of cards in Section V.
23-24	Number of statement cards in Section VI.
26-27	Number of pairs of DICTIONARY change cards in Section IV.
Section III	COMPUTED VARIABLE DEFINITION(S). One Definition is required for each computed variable being added to the HISTORY/KPATH file.
Columns	
1- 4	Variable name (i.e., C010)
6	'='
8-72	Variable definition
73-78	Format for the variable in F format right adjusted (i.e., F7.2). This format indicates variable size for printing purposes only (such as for program PRTVAR) and it is not used internally by any program.
Section IV	DICTIONARY CHANGE CARD(S). Two change cards are required for each Dictionary entry to be altered.

Card 1

Columns	
1- 4	Variable name to be changed, as it appears on the input.

Card 2

Columns	
1- 4	Variable name as it ought to appear on the output.
6	'='
8-72	Modified variable definition.
73-78	FORTRAN format for the variable. CXXX variables must be an "F" format and VXXX variables must be an "A" format. This format must be right adjusted in CC 78.

Section V HISTORY VARIABLE DEFINITION(S). Two cards are required for each History variable being added to the HISTORY/KPATH file.

Card 1

Columns	
1- 4	Variable name (i.e., V251)
7- 8	Number of computer words required to store the variable.

Card 2

Columns	
1- 4	Variable name (must be the same as Card 1)
6	'='
8-72	Variable definition
73-78	Format for the variable in A format, right adjusted (i.e., 1A3). This format is used internally by several programs and must be accurate.

Section VI — STATEMENT CARDS

Columns 23-24 of the control card contain the number of statement cards that follow (blank if none). If no statement cards are given, symbols (or dimensions) will be generated for all background variables. They will also be generated for all tasks, if /DISPERSE is used. However, it may be advisable for studies with a large number of tasks and a large number of background variables to state the tasks and background variables that will be used. Statement cards are of the following form:

cc 1-2	Number of symbols on this card (blank or zero, if none)
cc 3	Blank
cc 4-72	Up to 14 background variable names (i.e., V001) or task names (i.e., T001) separated by blanks. If a task exceeds T999, leave off the "T" and use 1000, 1001, etc.

A "1" in cc 24 of the PROGEN control card and a zero or blank in cc 1-2 of the statement card (i.e., a blank card will cause no symbols or dimensions to be generated for tasks or background variables.

NOTE: This card is NOT free format and must have the first name begin in cc 4 and must be exactly 4 characters long (A1,I3), followed by a blank. This card applies only to those task or background variable names used. All computed variable names will always be defined by program and user symbols will be generated by FORTRAN compiler.

Do NOT list variable ID of background (Vxxx) variables being generated in this run. It will cause multiple and incorrect definitions.

Section VII VERBS. Various HISTORY/KPATH file functions can be performed by using the following Verbs. Each Verb requires one card and must start in column 1. Verbs are optional except for /END which is required.

/OPEN	Reads and writes the Communication Region, Relative Location Table and Variable Dictionary. The HISTORY/KPATH file will be positioned to read and write case data after the /OPEN Verb is executed. When /OPEN is used, /CLOSE must also be used.
/READ	Reads a case data record. Case variables may be referenced as Cxxx for computed variables and Vxxx for History variables. A History variable longer than one computer word must be referenced with a subscript. Example: V105(1), V105(2).
/DISPERSE	Causes all the tasks for a given case to be unpacked into two arrays, "T" and "KT". Array "T" contains the percent time spent for each task in floating point format. Changing the contents of array "T" has no effect on the case record written out. Array "KT" contains the raw responses for each task. Each element of "KT" is equivalenced to an integer variable TXXXX where 'XXXX' is the task number. Changing the value of an element in "KT" or TXXXX will cause the percent time spent to be re-calculated when the /COLLECT is encountered.
/COLLECT	Will re-compute the percent time spend based upon the contents of array "KT". This may be used only if /DISPERSE is used.

NOTE: If /DISPERSE is used, but /COLLECT is not used, all task responses will be written out unchanged, regardless of modifications to array "KT".

/WRITE	Writes the input case data record along with any added variables which may have been generated by FORTRAN statements such as those described in Section VIII.
/AUDIT	Prints all computed and background variables for the first 5 cases and every 250th case following. It must follow the /WRITE and precede the /CLOSE cards.
/CLOSE	Reads and writes Duty/Task Titles and other records which follow case data records. /CLOSE is required if /OPEN has been used.
/END	This Verb is required and it must be at the end of all cards including FORTRAN statements which are described in Section VIII.

NOTE: The generated program will loop between (but not including) the /OPEN and the /CLOSE coding. When all case data has been read from the input file, the program will drop into the coding generated by the /CLOSE card.

Section VIII FORTRAN statements. FORTRAN statements may be mixed with Verbs. Any FORTRAN V statement is acceptable; however, there are some restrictions/rules unique to PROGEN which must be known to the user. These are detailed in the following paragraph.

GENERAL

1. FORTRAN statement numbers must not begin with '9'.
2. TYPE, DIMENSION, and DATA statements, if used, must appear at the beginning of all Section VII and VIII cards.

3. Verbs and FORTRAN statements may be mixed. It is the user's responsibility to insure that they are in a 'logically' correct sequence.

4. History variables (Vxxx) are alphanumeric, left adjusted with trailing blanks. Caution must be exercised when using data in this form. If possible use computed variables. History variables (Vxxx) have been declared 'INTEGER' by the program and must be used accordingly.

5. Computed variables (Cxxx) are binary floating point and are declared 'REAL' by the program and must be used accordingly. C001, C002, and C003 are special variables and declared integer.

6. No array or data name may be used if its second letter is C, V, N, T, S, or D.

7. The following are reserved areas which may be used by the user but not changed:

NCOMP	Number of new Computed variables
NVXXX	Number of new History variables
NSELCT	Number of cases in sample
NTASK	Number of tasks in sample
KCASE	Number of Dictionary records read
KCVARS	Number of Computed variables read
KVVARS	Number of words of History variables read
KTASKS	Number of non-zero tasks read
ENDOF	constant (ENDb̸OF)(Integer)
ISHF18	constant (2**18)

KC(I)	Array of computed variables
KV(I)	Array of History variables
T(I)	Array of percent time spent*
KT(I)	Array of Raw responses.* Same as T0001
KPATH	contains the number of cases read.

8. Variable names I thru N and S are used by Verbs and their contents will be destroyed during execution.

GENERAL DESCRIPTION

Cards are read and scanned for errors. The Communication Region, Relative Location Table and Variable Dictionary is read from the HISTORY/KPATH file. Study I.D. is verified, and certain values are computed for later use. The file is rewound. User supplied Verbs are translated into FORTRAN statements and those statements, along with user supplied FORTRAN statements, are stored into the source output element specified on the @Z.PROGEN card. The user must have provided an @FOR card to compile that element and an @XQT card to execute the program. The user must also provide a filenames card for the files used by his program. Any data cards read by the generated program must follow the filenames card. If "@Z.PROGEN ,PROGEN" is used for the processor call, an "@ADD Z.XQT-PROGEN" may be used to replace the @FOR, and @XQT cards described above.

NOTE:

The filenames card of the generated program must be set up as follows:

---

*These arrays are defined only if /DISPERSE is used.

1- 6	Any Program-ID desired.
8-18	Input HISTORY/KPATH filename. (FORTRAN Unit 10)
20-30	Output HISTORY/KPATH filename. (FORTRAN Unit 15) If none desired, must = 'SCRATCH' in 20-26.
32-42	Auxiliary Input filename 1 (FORTRAN Unit 11)
44-54	Auxiliary Input filename 2 (FORTRAN Unit 12)
56-66	Auxiliary Output filename 1 (FORTRAN Unit 16)
68-78	Auxiliary Output filename 2 (FORTRAN Unit 17)

FORTRAN Units 20 and 21 are automatic scratch files defined for the user.

PROGRAM IDENTIFICATION

Name	PRTVAR (Print Variables Report)
Language	FORTRAN V
Date Written	1966/April 1974
Programmer	CSC/Stacey/Weissmuller

FUNCTION

This program enables the user to select variables and print formats to produce a report of the case data values for the selected history and computed variables for selected groups of cases. The data can be sorted on the first variable printed. The sort may be in ascending, descending or tape order. Formerly called PRKPTH.

INPUT/OUT

Input	HISTORY/KPATH file.
	JOB DESCRIPTION file.
Output	A report will be produced on the printer and optionally added to the REPORT file.

CONTROL

Section I:	FILENAMES CARD
Columns	
1- 6	'PRTVAR'
8-18	HISTORY/KPATH file.
20-30	JOB DESCRIPTION file. (Optional)
32-42	REPORT file.
Section II:	IDENTIFICATION CARD
Columns	
1- 6	'PRTVAR'
8-15	Study ID

17	1 = Do not add the report to the REPORT file. 0 = Add the report to the REPORT file.

Section III: CONTROL CARD

Columns	
1	Number of PAGE HEADING CARDS in Section IV. 0-6 cards are allowed.
3	Number of pairs of COLUMN HEADING CARDS in Section V.
5- 6	Number of variables to print in Section VI.
8	Number of Format cards in Section VII.
10	SORT SEQUENCE. Blank = Descending sequence on the first variable printed. '1' = Ascending sequence on the first variable printed. '2' = Input order as the cases occur on the file. (No sort) '3' = Descending sequence on first and second variable. '4' = Descending on first but ascending on second variable. '5' = Ascending on first but descending on second variable. '6' = Ascending sequence on both first and second variable.
12-17	REPORT ID.
19-66	REPORT TITLE.

Section IV: PAGE HEADING CARDS

Columns	
1-80	Any information desired.

Section V: COLUMN HEADING CARDS

These cards must be in pairs, each pair representing a complete print line of 132 characters.

Columns	Card of the Pair	
1-66	1	The left-most 66 characters of a print line. First character is forms control.
1-66	2	The right-most 66 characters of a print line.

Section VI: VARIABLE SPECIFICATION CARDS

These cards contain the variables to print. 16 variables per card are allowed and as many cards as required may be used. Variables are printed in the order they appear on cards.

Columns	
1- 4	1st Variable.
6- 9	2nd Variable.
76-79	16th Variable.

Section VII: PRINT FORMAT SPECIFICATION

Columns	
1-72	These cards contain the FORTRAN print line formats for the selected variables. Instructions for format writing are found in the FORTRAN manual. The format characters required for a particular variable can be determined from the output of the 'PRDICT' program. The format must correspond in type and kind for each variable printed, but the print position across the page is completed determined by what the user punches on these cards. Format must be in parens. Print position one is always a carriage control character and it is recommended that this position be specified as 1X, which creates a blank and means single

space the report. The program also prints a sequence number in print positions 2-5, and a format must be included for this in the specification card. Therefore, it is recommended that all formats begin with (1X, I4,...).

NOTE: Either Section VIII or VIIIB may be used, but not both.

Section VIIIA: REQUEST CARD (Optional)

Columns	
1-80	This is a system produced request card. Only a single card may be used. If this section is used, the JOB DESCRIPTION file must be specified.

Section VIIIB: SPECIAL SAMPLE SELECTOR CARD (Optional)

Columns	
1-4	Variable ID (VXXX or CXXX) on which to select.
6	If variable above is: VXXX, then this is the number of six character fields to be used for exact match comparisons. A match on any of the specified fields will cause the case to be used. CXXX, then if this field is numeric, it represents an exact match value, however, if an "R" is in this column, the following 2-six character fields represent a low and high floating point range to be accepted.
8-13 14-19 ... 38-43	Match values must be left adjusted for background variable (VXXX) values, or right adjusted for computed variable values (CXXX).

Section IX: END CARD

Columns

1-3	END

NOTE: Multiple reports during a single execution of this program can be achieved by stacking Sections III through IX as many times as desired. Each set causes a new report to be printed.

GENERAL DESCRIPTION

The program reads the Section II control card and obtains the communication region from the HISTORY/KPATH file. Comparisons are made for correct study identification. If incorrect, appropriate messages are printed and the program halts. The program reads the relative location table, spaces over the dictionary records and reads Sections II through IX input cards. Processing the data records now begins. Each variable is selected from its respective portion of the data record and placed in a print array which is written to a SCRATCH file. The scratch is sorted and repassed. As each line is output, a page count is made and column page headings are printed when required.

MESSAGES

1. 'STUDY ID NO MATCH. TAPE IS XXXXXXXX.' Self explanatory.
2. 'NO VARIABLE SPECIFICATIONS, RUN DELETED.' CC 5=6 of the Section III card must not be blank.
3. 'NO FORMAT DATA PROVIDED, RUN TERMINATED.' CC 8 of the Section III card must not be blank.
4. 'AN INVALID VARIABLE HAS BEEN ENCOUNTERED ON A CONTROL CARD. IT IS XXXX.' An invalid variable has been specified in Section VI.

PROGRAM IDENTIFICATION

Name	REXALL (Inter-rater reliability)
Language	FORTRAN V
Date Written	1972/May 1973/Mar 1974
Programmer	Weissmuller

FUNCTION

This program computes and reports for a group of raters the average interrater reliability coefficient of a single rater and the stepped up reliability coefficient for the total group of raters. The program is most often used in conjunction with sets of task difficulty ratings made by a large number of supervisory personnel. The program computes and reports the distribution, mean, and standard deviation of the mean task ratings and the standard deviations of the ratings for each task.

INPUT/OUTPUT

Input	a. SETCHKed Rater file.
Output	a. Printed report.
	b. REPORT file.
	c. A deck of task means.

CONTROL

Section I FILENAMES CARD

Columns	
1- 6	REXALL
8-18	Filename of the SETCHKed Rater file
20-30	Filename of the REPORT file

Section II CONTROL CARD

Columns	
1- 6	REXALL
8	Number of PAGE HEADING cards in Section III. (b̸, 1, 2, or 3)
10-11	Number of format cards in Section IV. (01 to 50)
13	1 = Adjust ratings for all computations except rater correlations with raw means. Blank = Use raw ratings throughout the program.
15	1 = Suppress the printing of each rater's correlation with the means of all raters. 2 = Suppress both the correlation and distribution. Blank = Print the correlations and distribution.
17	1 = Punch a deck of NTASK means with a heading and format card. 'RR' will be punched for raw weights and 'RA' for adjusted weights. Means are based on raw ratings unless CC 13 = 1. Blank = Do not punch a deck.
19-66	REPORT TITLE.
68	Blank = Add the reliability coefficients and distribution, unless suppressed, on the REPORT file. 1 = Do not add the report to the REPORT file.
70-75	REPORT ID.
77-80	Study number to be punched on the weight cards if CC 17 = 1, or on the REQUEST Card if CC 68 is blank.

Section III PAGE HEADING CARDS (ƀ, 1, 2, or 3)

Columns	
1-80	Any alphanumeric information. This will be printed only on the reliability coefficients page.

Section IV RANGE CARD

Columns	
1	Number of digits in task response (1 to 6)
3- 8*	Largest acceptable task response (assumes 999999)
10-15*	Smallest acceptable task response (assumes 1)
17	"1" to print summary of task means and standard deviations. Blank to not print the summary.
19	"1" if case deletion cards follow. Blank if, Section V, case deletions, is not used.
21-26	Scale factor for task range above. This value will be divided into all task responses which fall within the range in CC 3-8 and 10-15. This field must have a decimal point punched. If blank, scale will be set to 1.0.

*These values must be NDIGITS long, where NDIGIT is the value in CC 1. If NDIGIT is 3 a low value of 5 must be written 005 left adjusted in CC 10.

Section V CASE DELETIONS

Part A Case Identification

Columns	
1- 4	Case-ID of case to be bypassed.
6- 9	Case-ID of case to be bypassed.
...	
76-79	Case-ID of case to be bypassed

Part A may be repeated as often as necessary, with an upper limit of 1000 case deletions.. The Case-ID's do not have to be in any particular order. and the cards do not have to be full before starting a new card.

Part B Stop Card

Columns	
1- 4	STOP

The Part B card must follow the last case identification card and immediately precede the first format card.

Section VI FORMAT CARDS (01 to 50)

Columns	
1- 4	C...
5-73	'T' in the relative location of the case record in which a task is to be used. Note that task responses may be 1 to 6 digits long, and the 'T' locates the left most digit of the task response. 0 is taken as a non-response. All characters other than 'T' will be ignored.
74-79	Check field. The contents of this field must match the same field on the corresponding card for each rater. Normally columns 74 and 75 contain a card number, while columns 76-79 contain the Study ID.

NOTE: The number of format cards must be equal to the number of cards required for a single rater. In addition, the format cards must be input in the same sequence (based on card number in CC 74-75) as the rater cards. Any mismatch on the check field for any rater will cause an automatic abort.

Section VII END CARD

Columns

1- 3 END

NOTE: SECTIONS II through VII may be repeated for stacked runs on the same data file.

GENERAL DESCRIPTION

The input file is passed, specified cases are deleted, the responses are scaled, and task sums are recorded in core, while each raters responses, are all written on disk. Any mismatch on a check field for any rater causes the report to be skipped, and the input file to be rewound. For this reason, the input file should come directly from SETCHK. The recorded task sums for all selected raters are used to compute mean ratings for each task. If the ratings are to be adjust, the task sums (not the means) are cleared to zero. The disk is passed and each rater's correlation with the means of all raters is computed. If the scores are to be adjusted, task sums and sums of squares are re-accumulated. Unless suppressed, the correlations are printed as the disk is passed. The data in core is used to calculate task means, standard deviations, and the distributions of these statistics. If requested, a means deck is punched and the summary is printed. Next, the reliability coefficients are computed and printed along with the data used to calculate them. Unless suppressed, the distributions are printed. If specified, the report is added to the REPORT file, and the REQUEST card is punched. The program is reset and attempts to read another control card.

MESSAGES

1. 'SKIPPING TO END CARD DUE TO ABOVE ERROR.' Self-explanatory
2. 'NUMBER OF TITLE CARDS IS NOT IN THE RANGE 0-3.' Card Column 8 of the Section II card must be b̸, 0, 1, 2, or 3.
3. 'NUMBER OF FORMAT CARDS IS NOT CORRECT.' The number in CC 10-11 of the Section II card must be right adjusted and must be in the range 1 to 50, and must be the number of format cards actually present. If the number of PAGE HEADING cards in CC 8 is not correct, it may appear as this error since a format card may mistakenly be read as a PAGE HEADING card. Also, the END card must follow the last format card and this error will occur if the END card is not present or out of sequence.
4. 'C... IS NOT IN CC 1-4 OF THIS FORMAT CARD.' This message will appear if the format card is mispunched, or if the number of PAGE HEADING cards is incorrect.
5. 'CASE ID MISMATCH. RUN ABORTS.' The case ID on the first card of each rater must match every ID on every card for this rater. This error will occur if the number of format cards is not equal to the number of cards for each rater. An additional line will be printed to locate the error.
6. 'CHECK CODE DOES NOT MATCH. RUN ABORTS.' The location of the error is printed on the line following the above message. See SECTION VI - FORMAT CARDS for an explanation.
7. 'UNEXPECTED END OF FILE ON CARD READER.' At least the last END card, if not more cards, are missing from the last set of cards.
8. 'END OF REXALL.' All sets of cards have been processed or bypassed and the program halts.

PROGRAM NOTES

The general formula for inter-rater reliability coefficients is

RXX = (BTMS - WMS)/(BTMS + (K/X - 1)*WMS)

where

BTMS = between task mean squares
WMS = within task mean squares
K = average number of raters per task on which this sample is based
X = the number of raters to which this formula is being projected.

Hence, R11, the reliability coefficient for a single rater (X = 1) becomes:

R11 = (BTMS - WMS)/(BTMS + (K - 1)*WMS)

Also, RKK, the reliability coefficient for this sample of K raters (X = K) is:

RKK = (BTMS - WMS)/BTMS

If all raters rate all tasks, the average number of raters per task, K, will be NRATER. In general, however,

K = (1/XTASK) (NRESP - (NRESQ/NRESP))

where

XTASK = the number of tasks rated minus 1
NRESP = the total number of ratings for all raters, and
NRESQ = the sum of the squares of the number of ratings for each task.

If the ratings are to be adjusted, each rating r, for every rater x, is altered to yield the new rating r_A thusly:

$$r_A = (SD_c/SD_x)\ (r - r_x) + Y_x + CMN - GMN$$

where

r_A = the adjusted rating

SD_c = the control standard deviation = 1.0

SD_x = the rater's standard deviation

r = the raw rating (in range 1 to 9)

r_x = the rater's mean rating

Y_x = the unweighted mean of all raters rating the same tasks as rater X.

CMN = the control mean = 5.0

GMN = the unweighted grand mean for all raters on all tasks

Note that SD_c, CMN, and GMN are constant for all raters.

The following notes explain terms used with the individual rater's correlations:

The sample mean is the mean of the mean ratings by all raters on those tasks which were rated by this individual rater. See Y_x above.

The T-Value is computed thusly:

T = CORR * SQRT (X - 2.0)/SQRT (1. - CSQ)

where

[illegible]RR = the rater's correlation

CSQ = CORR * CORR

X = number of tasks rated by this rater

SQRT = the square root function

If X is less than 2, T is set to 0.

If CORR is equal to 1, T is set to 99999.0

The primary reference materials for the inter-rater reliability coefficient formulations are:

Haggard, Ernest A. Intraclass Correlation and the Analysis of Variance, p. 14 and p. 89

Lindquist, E. F. Design and Analysis of Experiments in Psychology and Education, p. 361.

PROGRAM IDENTIFICATION

Name	SETCHK (Set Check)
Language	FORTRAN V
Date Written	Apr 1973
Programmer	Barton

FUNCTION

This program edits the Raw Data file which will become input to program INPSTD. Valid cases are written on the SETCHKed raw data file and invalid cases are written on the Invalid Case file. SETCHK will accept 20000 cases, 1700 task variables per case, 1700 history variables per case and 26 duty variables. The Raw Data file must be ordered Card number within Case number prior to the SETCHK program.

INPUT/OUTPUT

Input	a. Raw Data file.
	b. Control card.
	c. CODAP format card deck.
Output	a. SETCHKed raw data file.
	b. Invalid Case file.
	c. List of error messages for invalid cases.

CONTROL

Section I: FILENAMES CARD

Columns	
1- 6	"SETCHK"
8-18	Filename of the Raw Data file.
20-30	Filename for the SETCHKed raw data file.
32-42	Filename for the Invalid Case file.

Section II: CONTROL CARD

Columns	
1- 6	"SETSTD"
7-12	MAXTSK. Maximum acceptable value for a task response right adjusted. Maximum is 999999.
14-15	NCARD. Number of cards in a case right adjusted. Maximum is 50.
17-20	NTASK. Number of tasks in the inventory right adjusted. Maximum is 1700.
22-25	NHIST. Number of history variables in the inventory right adjusted. Maximum is 1700 (1700 computer words are reserved for history variables; therefore, 1700 variables are allowed only if all variables are 6 characters or less in size).
27-28	NDUTY. Number of duties in the inventory right adjusted. Maximum is 26.

Section III: FORMAT CARDS

See 'FORMAT CARDS' in this manual.

GENERAL DESCRIPTION

SETCHK first varifies that all control card parameters are correct. Any out-of-range values, including non-numeric characters and/or trailing blanks, will cause the run to abort. The CODAP format cards are processed by the subroutine FMTGEN. Checks are made for consistency between values generated by FMTGEN and those specified by the control card; the run aborts if these checks fail. As each case is read from the Raw Data file, tests are made for proper number of cards, check character match ups, and at least one task value in range. Any discrepancy will cause that case to be deleted from the SETCHKed raw data file and entered in the Invalid Case file; an appropriate message is printed. As out-of-range and/or blank task values are encountered, they are reset to zero. Upon completion of processing, numbers of cases accepted and deleted, and percent deleted, are printed, as are the numbers of cases, tasks per case, duties per case, and history variables per case which reside on the SETCHKed raw data file.

MESSAGES

1. 'XXXX HISTORY VARIABLES WERE DEFINED BY FORMAT CARDS AND XXXX HISTORY VARIABLES WERE INDICATED ON THE CONTROL CARD. RUN ABORTED.' Self explanatory.
2. 'XXXX TASK VARIABLES WERE DEFINED BY FORMAT CARDS AND XXXX TASK VARIABLES WERE INDICATED ON THE CONTROL CARD. RUN ABORTED.' Self explanatory.
3. 'XXXXX COMPUTER WORDS ARE REQUIRED TO STORE HISTORY VARIABLES. THIS EXCEEDS THE MAXIMUM OF 1700. RUN ABORTED.' Reduce the number of history variables until the 1700 word restriction is satisfied.
4. 'TOO MANY CARDS IN CASE XXXX. XXXXXX CARDS FOUND. CASE DELETED.' Self explanatory.
5. 'TOO FEW CARDS IN CASE XXXX. ONLY XX CARD(S) FOUND. CASE DELETED.' Self explanatory.
6. 'CHECK CHARACTER NO MATCH. CASE DELETED. CASE CARD IMAGE IS XXXX---XXXX.' Check characters used in format cards as described in Section III must match characters in corresponding columns of the Raw Data file.

7. 'XX OUT OF RANGE TASK VALUE(S) SET TO ZERO. CARD IMAGE IS XXXX-4-XXXX.' Values set to zero on SETCHKed Raw Data file.

8. 'CASE XXXX HAD NO TASKS WITHIN RANGE. CASE DELETED.' Self explanatory.

9. 'UNEXPECTED END OF FILE IN CONTROL/FORMAT DECK. RUN ABORTED.' NCARD on control card is greater than actual number of format cards.

10. 'THE MAXIMUM ACCEPTABLE VALUE FOR A TASK, SPECIFIED BY CONTROL CARD, IS INCORRECT. RUN ABORTED.' MAXTSK may not exceed 999999, nor be less than 1.

11. 'THE NUMBER OF CARDS PER CASE SPECIFIED BY CONTROL CARD IS INCORRECT. RUN ABORTED.' NCARD may not exceed 50, nor be less than 1.

12. 'THE NUMBER OF TASKS SPECIFIED BY CONTROL CARD IS INCORRECT. RUN ABORTED.' NTASK may not exceed 1700, nor be less than 1.

13. 'THE NUMBER OF HISTORY VARIABLES SPECIFIED BY CONTROL CARD IS INCORRECT. RUN ABORTED.' NHIST may not exceed 1700, nor be less than 1.

14. 'THE NUMBER OF DUTIES SPECIFIED BY CONTROL CARD IS INCORRECT. RUN ABORTED.' NDUTY may not exceed 26, nor be less than 1.

15. 'FORMAT CONTROL CARD DOES NOT BEGIN C... RUN ABORTED.' Format cards are not in their proper position in the run stream, or they are incorrectly prepared.

16. 'A FORMAT CARD DOT (.) WAS USED INCORRECTLY. RUN ABORTED.' Read Section III of the write-up.

17. 'THE NUMBER OF FORMAT CARD CHECK CHARACTERS EXCEEDS 500. RUN ABORTED.' Remove as many format characters as required to satisfy the 500 character limit.

18. 'THE NUMBER OF TASK VARIABLES DEFINED BY FORMAT CARDS EXCEEDS 1700. RUN ABORTED.' 1700 task variables is maximum.

19. 'THE NUMBER OF HISTORY VARIABLES DEFINED BY FORMAT CARDS EXCEEDS 1700. RUN ABORTED.' 1700 history variables is maximum.

20. 'THE FORTRAN FORMAT GENERATED FROM FORMAT CARDS EXCEEDS ALLOCATED STORAGE. SIMPLIFY INPUT. RUN ABORTED.' 1000 words are reserved to store the FORTRAN format. It has been exceeded.

21. 'MORE THAN ONE TASK LENGTH ON FORMAT CARDS. RUN ABORTED.' The number of dots (.) following a "T" must remain constant for all tasks.

22. 'SETCHK COMPLETE

```
XXXXXX CASES ACCEPTED
XXXXXX CASES DELETED
XXXX.X PERCENT CASES DELETED
  EDITED RAW DATA TAPE CONTAINS -
XXXXXX CASES
XXXXXX TASKS
XXXXXX DUTIES
XXXXXX HISTORY VARIABLES'
```

Successful run execution.

PROGRAM IDENTIFICATION

Name	TSETUP (Tape Initialization)
Language	FORTRAN V
Date Written	May 73
Programmer	CSC/Weissmuller

FUNCTION

Initialize 2 files which will become the REPORT file and JOB DESCRIPTION file. Other programs may then add reports and statistical data to these files. The initializing action consists of (1) writing 'BEGIN REPORT' sentinel and an EOF mark on the REPORT file, and (2) writing an EOF mark on the JOB DESCRIPTION file.

INPUT/OUTPUT

Input	a. FILENAMES card
Output	a. JOB DESCRIPTION file
	b. REPORT file

CONTROL

Section I — FILENAMES CARD

Columns	
1- 6	TSETUP
8-18	Filename for the JOB DESCRIPTION file.
20-30	Filename for the REPORT file.
33-36	Last Report on previous Report File. Right adjusted. Blank if this is the first Report File for this study.

GENERAL DESCRIPTION

The program reads the FILENAMES card and initializes the JOB DESCRIPTION file with an EOF mark if filename is given in cc 8-18 and initializes the REPORT file with a 'BEGIN REPORT' sentinel and and EOF mark if filename is given in cc 20-30. The 'BEGIN REPORT' sentinel consists of a three word record; the two words 'BEGIN REPORT' and a binary integer value. The binary integer value is a unique identification number for the report following the sentinel. There is always one more sentinel on the file than there are reports because the last record on the file is always a 'BEGIN REPORT' sentinel for the next report to be added. Because a multi-reel report file would confuse the 3-copy cycling routines (CYCLES), when a report file completely fills a single reel, a new 3-copy set must be created with a different filename. To keep the report numbers on the 3-copy set in phase with the Job Description file, the last good report number from the first 3-copy set must be entered right adjusted in cc 33-36 above. Note: A blank in those columns indicates this is the first 3-copy set made and will generate a 'BEGIN REPORT' sentinel with a value of 1. A file will not be initialized if filename is not given.

PROGRAM IDENTIFICATION

Name	TSKGRP (Task Groups)
Language	FORTRAN V
Date Written	Aug 74
Programmer	Weissmuller

FUNCTION

To compute and print the following: (1) Task statements for selected tasks performed by a selected group of members, (2) Task sequence number, (3) Task index (mean), (4) Standard Deviation of task ratings, (5) number of members performing, (6) Percent of members performing, (7) Average percent time spent by members performing, (8) Average percent time spent by all members, (9) Cumulative sum of average percent time spent by all members.

INPUT/OUTPUT

Input
- a. HISTORY file or KPATH file.
- b. Deck or tape containing task ratings, or deck of mean task difficulty ratings.
- c. Control cards.

Output
- a. Printed report.
- b. Full report on the REPORT file if requested.
- c. Mean task difficulty rating deck if requested.

CONTROL

Section I: FILENAMES CARD

Columns	
1-6	'TSKGRP'
8-18	Filename of the HISTORY or KPATH file.

CONTROL (continued)

Section I:

20-30	Filename of the SETCHKED RATING file.
32-42	Filename of the REPORT file

Section II:

Columns	
1-6	TSKGRP
8-15	Study ID on the KPATH or HISTORY file.
17	1 = input is task ratings from a SETCHKED file. Rating data must be in a format as follows: case control number in cc 1-4, task ratings in cc 5-73, and card sequence number in cc 74-75. Background data may be present, but will be ignored. 2 = input in task ratings from cards. The format of these cards must be as described in Option 1 above. 3 = input is 'mean' task values from cards. There must be NTASK values in duty/task order punched in a format specified by a card which directly precedes the 'means' deck. CC 1-8 of each card must contain control information as described in the following option.

CONTROL (continued)

Section II

Columns	
19	1 = Punch a deck of mean task values. This deck will contain mean values in duty/task order. NTASK values will be punched 12 to a card in F6.2 format. The first 8 columns of each card contain the following control information: Card sequence number in cc 1-2, 'TX' in cc 3-4, and the first 4 characters of study I.D. in cc 5-8. 0 = Do not punch a deck.
21-22	Number of cards per case in the rating data. Not required if column 17 equals 3.
24	The minimum rating allowable for any task. Assumes 1 if blank. Not required if column 17 equals 3.
26	The maximum rating allowable for any task. Assumes 7 if blank. Not required if column 17 equals 3.
28-31	Number of cases (raters) in the rating data. Not required if column 17 equals 3.

Section III: Format cards. If column 17 of the Section II card equals 1 or 2, format cards are required to describe the task rating data. See 'FORMAT CARDS' in this manual.

Section IV: Task index rating deck. If column 17 of the Section II card equals 2, insert the deck which was described by the format cards in the previous section.

CONTROL (continued)

Section V: Task rating 'Means' Deck. If column 17 of the Section II card equals 3, insert a deck containing NTASK values in duty/task order punched in a format specified on a card directly preceding the means deck. Means are usually punched 12 to a card in F6.2 format, so the format card would be punched (I2,A2,A4,12F6.2) in columns 1-17. CC 1-2 of these cards will be sequence checked by the program.

Section VI: CONTROL CARD

Columns	
4	Sort option for printed output 1 = Ascending on task index 2 = Descending on task index 3 = Duty/Task order 4 = Ascending on percent members performing 5 = Descending on percent members performing
6	Specify the number of PAGE HEADING cards following. Must be between 0 and 6.
8	0 = Add this report to the REPORT file. 1 = Do not add this report to the REPORT file.

CONTROL (continued)

Section VI:

Columns	
10	Print option 1 = Option 1 print. Short print. Option 1 is selected automatically by the program if column 17 of the Section II card is 3. Items numbered 2, 3, 5, 6, 7 which are listed under program 'FUNCTION' will be printed when option '1' is selected. 2 = Option 2 print. Long print. Option 2 can be used only when column 17 of the Section II card is 1 or 2. All items listed under program 'FUNCTION' will be printed when this option is selected.
12-17	REPORT ID.
19-66	REPORT TITLE.

Section VII: TASK GROUP CARDS

Columns	
1- 4	Task number to be included in this group (right adjusted)
5- 8	Task number to be included in this group (right adjusted)
9-12	Task number to be included in this group (right adjusted)
...	(Continue 4 digit field as above)
77-80	Task number to be included in this group (right adjusted)
NOTE:	Enter "STOP" after the last task. (e.g. "177STOP")

CONTROL (continued)

Section VIII: PAGE HEADING CARDS

Columns

1-80	Any information the user desires. PAGE HEADING cards appear on the first page of the report. There must be between 1 and 6 PAGE HEADING cards.

Section IX: SAMPLE SELECTOR CARDS

See 'SAMPLE SELECTION' in this manual following program writeups.

NOTE: Sections VI-IX may be repeated for multi-report runs.

GENERAL DESCRIPTION

The program reads the communications region from the KPATH or HISTORY file and computes the number of reports that it can generate with one pass of the file. It then reads ratings (either ratings, or mean values) from card or from tape. If ratings were read, the program computes means and standard deviations for the ratings. Control cards are read until EOF is encountered or until IMAX 'sets' have been read. Data cases are read and tested to see if they fall within the range specified by selector variables for any of the 'sets'. The percent time spent values for selected cases are summed and stored for later computations. When the input has been passed, various values are computed, and the report is printed and added to the REPORT file if requested. If the program has encountered EOF it terminates. If not, it recycles.

MESSAGES

1. 'STUDY I.D. NO MATCH. RUN TERMINATED.'

   The study I.D. punched in the Section II control card does not match the study I.D. on the HISTORY or KPATH files.

2. 'COLUMN XX OF I.D. CARD IN ERROR. RUN TERMINATED.'

   Read Section II, column 17 and 24.

3. 'CASE XXXX HAS RATING XXX OUT OF RANGE. RATING SKIPPED.'

   Response was greater than allowable as specified in Section II, column 26.

4. 'THE NUMBER OF T'S IN FORMAT CARDS DO NOT EQUAL NTASK FROM COMM REGION. RUN TERMINATED.'

   NTASK from the Communications Region of the HISTORY or KPATH file does not agree with 'T' count.

5. 'ERROR IN CASE XXXX OF RATER DECK. RUN ABORTED.'

   Either a card is missing or out of sequence, or case control number or card number is mispunched.

6. 'INCORRECT SORT SEQUENCE FOR XXXXXX. ASSUMED 1.'

   Read Section VI, column 4.

7. 'THE NUMBER OF TITLE CARDS SPECIFIED FOR XXXXXX IN ERROR. RUN TERMINATED.'

   Read Section VI, column 6.

8. 'PRINT OPTION INCORRECTLY SPECIFIED FOR XXXXXX. ASSUMED OPTION 2.'

   Read Section VI, column 10.

9. 'REPORT FILE OPTION INCORRECTLY SPECIFIED FOR XXXXXX. ASSUMED 0.'

   Read Section VI, column 8.

10. 'ERROR IN VARIABLE INTERACTION OR DEFINITION CARD FOR XXXXXX. REPORT SKIPPED.'

    Read the description 'SAMPLE SELECTION.'

11. 'SPECIFIED VARIABLE LENGTH FOR XXXXXX IN ERROR. REPORT SKIPPED.'

    Read the description 'SAMPLE SELECTION.'

PROGRAM IDENTIFICATION

Name	TSKNDX (Task Index)
Language	FORTRAN V
Date Written	1970/May 1973
Programmer	Stacey

FUNCTION

To compute and print the following: (1) Task statements for tasks performed by a selected group of members, (2) Task sequence number, (3) Task index (mean), (4) Standard Deviation of task ratings, (5) Number of members performing, (6) Percent of members performing, (7) Average percent time spent by members performing, (8) Average percent time spent by all members, (9) Cumulative sum of average percent time spent by all members.

INPUT/OUTPUT

Input
- a. HISTORY file or KPATH file.
- b. Deck or tape containing task ratings, or deck of mean task difficulty ratings.
- c. Control cards.

Output
- a. Printed report.
- b. Full report on the REPORT file if requested.
- c. Mean task difficulty rating deck if requested.

CONTROL

Section I: FILENAMES CARD

Columns	
1-6	'TSKNDX'
8-18	Filename of the HISTORY or KPATH file.

CONTROL (continued)

Section I:

20-30	Filename of the SETCHKED RATING file.
32-42	Filename of the REPORT file.

Section II: IDENTIFICATION

Columns	
1-6	TSKNDX
8-15	Study ID on the KPATH or HISTORY file.
17	1 = input is task ratings from a SETCHKED file. Rating data must be in a format as follows: case control number in cc 1-4, task ratings in cc 5-73, and card sequence number in cc 74-75. Background data may be present, but will be ignored. 2 = input in task ratings from cards. The format of these cards must be as described in Option 1 above. 3 = input is 'mean' task values from cards. There must be NTASK values in duty/task order punched in a format specified by a card which directly precedes the 'means' deck. CC 1-8 of each card must contain control information as described in the following option.

CONTROL (continued)

Section II

Columns

19	1 = Punch a deck of mean task values. This deck will contain mean values in duty/task order. NTASK values will be punched 12 to a card in F6.2 format. The first 8 columns of each card contain the following control information: Card sequence number in cc 1-2, 'TX' in cc 3-4, and the first 4 characters of study I.D. in cc 5-8. 0 = Do not punch a deck.
21-22	Number of cards per case in the rating data. Not required if column 17 equals 3.
24	The minimum rating allowable for any task. Assumes 1 if blank. Not required if column 17 equals 3.
26	The maximum rating allowable for any task. Assumes 7 if blank. Not required if column 17 equals 3.
28-31	Number of cases (raters) in the rating data. Not required if column 17 equals 3.

Section III: Format cards. If column 17 of the Section II card equals 1 or 2, format cards are required to describe the task rating data. See 'FORMAT CARDS' in this manual.

CONTROL (continued)

Section IV: Task index rating deck. If column 17 of the Section II card equals 2, insert the deck which was described by the format cards in the previous section.

Section V: Task rating 'Means' Deck. If column 17 of the Section II card equals 3, insert a deck containing NTASK values in duty/task order punched in a format specified on a card directly preceding the means deck. Means are usually punched 12 to a card in F6.2 format, so the format card would be punched (I2,A2,A4,12F6.2) in columns 1-17. CC 1-2 of these cards will be sequence checked by the program.

Section VI: CONTROL CARD

Columns	
4	Sort option for printed output 1 = Ascending on task index 2 = Descending on task index 3 = Duty/Task order
6	Specify the number of PAGE HEADING cards following. Must be between 1 and 6.
8	0 = Add this report to the REPORT file. 1 = Do not add this report to the REPORT file.

CONTROL (continued)

Section VI:

Columns	
10	Print option 1 = Option 1 print. Short print. Option 1 is selected automatically by the program if column 17 of the Section II card is 3. Items numbered 2, 3, 5, 6, 7 which are listed under program 'FUNCTION' will be printed when option '1' is selected. 2 = Option 2 print. Long print. Option 2 can be used only when column 17 of the Section II card is 1 or 2. All items listed under program 'FUNCTION' will be printed when this option is selected.
12-17	REPORT ID.
19-66	REPORT TITLE.

Section VII: PAGE HEADING CARDS

Columns	
1-80	Any information the user desires. PAGE HEADING cards appear on the first page of the report. There must be between 1 and 6 PAGE HEADING cards.

CONTROL (continued)

Section VIII: SAMPLE SELECTOR CARDS

See 'SAMPLE SELECTION' in this manual following program writeups.

Note: Sections VI, VII, and VIII may be repeated for multi-report runs.

GENERAL DESCRIPTION

The program reads the communications region from the KPATH or HISTORY file and computes the number of reports that it can generate with one pass of the file. It then reads ratings (either ratings, or mean values) from card or from tape. If ratings were read, the program computes means and standard deviations for the ratings. Control cards are read until EOF is encountered or until IMAX 'sets' have been read. Data cases are read and tested to see if they fall within the range specified by selector variables for any of the 'sets.' The percent time spent values for selected cases are summed and stored for later computations. When the input has been passed, various values are computed, and the report is printed and added to the REPORT file if requested. If the program has encountered EOF it terminates. If not, it recycles.

MESSAGES

1. 'STUDY I.D. NO MATCH. RUN TERMINATED.'

   The study I.D. punched in the Section II control card does not match the study I.D. on the HISTORY or KPATH files.

MESSAGES (continued)

2. 'COLUMN XX OF I.D. CARD IN ERROR. RUN TERMINATED.'

   Read Section II, column 17 and 24.

3. 'CASE XXXX HAS RATING XXX OUT OF RANGE. RATING SKIPPED.'

   Response was greater than allowable as specified in Section II, column 26.

4. 'THE NUMBER OF T'S IN FORMAT CARDS DO NOT EQUAL NTASK FROM COMM REGION. RUN TERMINATED.'

   NTASK from the Communications Region of the HISTORY or KPATH file does not agree with 'T' count.

5. 'ERROR IN CASE XXXX OF RATER DECK. RUN ABORTED.'

   Either a card is missing or out of sequence, or case control number or card number is mispunched.

6. 'INCORRECT SORT SEQUENCE FOR XXXXXX. ASSUMED 1.'

   Read Section VI, column 4.

7. 'THE NUMBER OF TITLE CARDS SPECIFIED FOR XXXXXX IN ERROR. RUN TERMINATED.' Read Section VI, column 6.

8. 'PRINT OPTION INCORRECTLY SPECIFIED FOR XXXXXX. ASSUMED OPTION 2.'

   Read Section VI, column 10.

9. 'REPORT FILE OPTION INCORRECTLY SPECIFIED FOR XXXXXX. ASSUMED 0.'

   Read Section VI, column 8.

10. 'ERROR IN VARIABLE INTERACTION OR DEFINITION CARD FOR XXXXXX. REPORT SKIPPED.'

    Read the description 'SAMPLE SELECTION.'

11. 'SPECIFIED VARIABLE LENGTH FOR XXXXXX IN ERROR. REPORT SKIPPED.'

Read the description 'SAMPLE SELECTION.'

PROGRAM IDENTIFICATION

Name	VARGEN (Variable Generator)
Language	FORTRAN V
Date Written	1967/May 1973
Programmer	CSC/Stacey

FUNCTION

This program calculates new computed variables by applying input data to the task values of each case. The task values may be time spent percentages or "do - don't do" values (1,0). The input data consists of a vector of weights, a scaling (standardizing) factor and a specified calculation formula (four options). A newly created variable is given a variable identification number and is added into the computed variable portion of the case data records on the HISTORY or KPATH files.

INPUT/OUTPUT

Input	HISTORY/KPATH file
Output	New HISTORY/KPATH file containing newly formed Computed Variables.

CONTROL

Section I FILENAMES CARD

Columns	
1- 6	VARGEN
8-18	Filename of HISTORY/KPATH file.
20-30	Filename for new HISTORY/KPATH file.

Section II IDENTIFICATION CARD

Columns	
1- 6	VARGEN
8-15	Study identification, eight alphanumeric characters.
17-18	Number of Computed Variables being added.

Section III DICTIONARY CARDS

Columns	
1-80	Computed Variable title, beginning with 'Cxxxb̸=b̸'. One card is required for each new variable formed. The titles will be copied to the New HISTORY file in the same order that they are read from cards. The program does not resequence them. Each title must have a FORTRAN format right adjusted in cc 73-78.

Section IV WEIGHT CARDS

This section consists of three types of cards. The section can be repeated as often as required - each group of cards describing one set of weights that is to be applied to task values to form Computed Variable values.

Part (a)

Columns	
1- 6	Weight set identification name (any six characters)
8-17	Scale factor for this weight set, w/decimal point.

Part (b)

Columns

1-72 FORTRAN format by which each card of a weight set should be read. The format must begin with an open parenthesis and terminate with a closed parenthesis. The format must yield only floating point values. Legal FORTRAN edit codes are X, F, and E. Codes A, H, and I must not be used. CC 1-12 must be punched as follows: (I2,A2,A4,NN) where NN is the number of weights contained in each card. A complete format could be (I2,A2,A4,72F1.0).

Part (c)

Columns

1-80 Weight cards corresponding in position, number and type to previous formats. There must be NTASK weights, where NTASK is the number of tasks in this study. The first 8 columns must be control information as follows: cc 1-2 must be a sequence number which will be checked by the program, cc 3-4 should contain a code to identify the source of the weights such as 'AV' to indicate that the source was program AVALUE, and cc 5-8 should be a study number. Columns 3-8 are for visual use only and are not checked by the program.

Section V END CARD

Columns

1- 3 END

Section VI OPTION CARD

This section selects the computation method and set of weights by name that will be applied to the task variables of the case data to generate each new Computed Variable.

Columns	
1- 4	CXXX, where XXX is the new Computed Variable number, which must be greater than the number of computed variables already on the input HISTORY/KPATH file.
6	Option number, 1, 2, 3, or 4. The available options are:

Option 1

$$C_N = \frac{1}{S} \sum_{1}^{P} W_i T_i$$

Option 2

$$C_N = \sum_{1}^{P} W_i D_i$$

Option 3

$$C_N = \sum_{1}^{P} \min (W_i, T_i)$$

Option 4

$$C_N = \frac{1}{M} \sum_{1}^{P} W_i D_i$$

Where:

C_N = the new Computed Variable being formed.

W_i = an array of NTASK weights.

T_i = an array of NTASK task values for a given data case. The T_i values are percentages in floating point and input as XXX.XXX.

S = a scale factor to position the decimal point.

$D_i = 1$ if $T_i \neq 0$

$D_i = 0$ if $T_i = 0$

M = the number of non-zero task responses, that is the number of $T_i \neq 0$.

P = the number of tasks in the study. Hence P = NTASK

8-13 Weight set's name; must match one of the names on a part (a) card in Section IV.

There must be as many cards in Section VI as there are variables to be formed. These cards may be read in any order. They need not be in Computed Variable number sequence, as the program will sort them.

GENERAL DESCRIPTION

Program VARGEN reads card input data describing Computed Variables titles, weights, and the manner in which the weights are to be used. The program then copies the input HISTORY/KPATH file onto the output HISTORY/KPATH file, inserting the new titles, computing and adding new Computed Variables to each data record.

MESSAGES

1. 'DATA REQUIRES TOO MUCH STORAGE. PROCESS THE DATA IN TWO OR MORE RUNS.' Core storage limitations prevent all the new computed variable formations in one run.

2. 'NO. OF COMPUTED VARIABLES PLUS NO. OF HISTORY VARIABLES = xxxx. RUN ABORTED.' The total number of variables exceeds the maximum limit of 928.
3. 'COMPUTED VARIABLE NUMBER = xxxx IS OUT OF RANGE. RUN ABORTED.' Either the computed variable number is not greater than the previous highest number of computed variables on the input file, or it is greater than the sum of the number of computed variables on the input file plus the number that are presently being formed.
4. 'OPTION CARDS xxxx AND xxxx SPECIFY THE SAME COMPUTED VARIABLE NUMBER. RUN ABORTED.' Duplicate or invalid option cards.
5. 'WEIGHT SET NAME xxxxxx ON AN OPTION CARD DOES NOT MATCH ANY OF THE INPUT WEIGHT SET I.D. NAMES. RUN ABORTED.' The indicated weight set name was in Column 8-13 on an option card but did not appear in Columns 1-6 on any of the weight cards in Section IV.
6. 'WEIGHT DECK OUT OF SEQUENCE. RUN ABORTED.' Weight cards must be in sequence on cc 1-2.

PROGRAM IDENTIFICATION

Name	VARMEN (Variable Means and Standard Deviations)
Language	FORTRAN V
Date Written	July 1974
Programmer	Weissmuller

FUNCTION

This program extracts up to 100 computed and background variables from a CODAP KPATH or HISTORY file and computes means and standard deviations. The program computes and prints the number of valid and invalid cases in the sample and means and standard deviations of variables. A subsample may be selected for correlation according to the value (0 or 1) of a computed variable.

INPUT/OUTPUT

Input	a. HISTORY file or KPATH file.
	b. Control cards.
Output	a. Printed output.

CONTROL

Section I: FILENAMES CARD

Columns	
1- 6	VARMEN
8-18	Filename of the HISTORY or KPATH file.

Section II: IDENTIFICATION

Columns	
1- 6	VARMEN
8-15	Study I.D. on the HISTORY or KPATH file.
17	Number of PAGE HEADING CARDS following. May be any number 0 through 9.

Section III: PAGE HEADING CARDS

Columns	
1-80	Any information the user desires. PAGE HEADING CARDS are not required.. A maximum of 9 cards may be used.

Section IV: CONTROL CARD

Columns	
1- 3	The number of variables punched in the cards that follow. Maximum is 100.
5- 8	Subsample selector variable. If the total sample is desired, leave this field blank. If specified, this variable must be a computed variable (Cxxx) which has been set to 0 or 1 by a prior run.
10	If a selector variable has been specified, this field must equal 0 or 1. Specify 0 to select cases only when the selector variable equals 0 and specify 1 to select cases only when the selector variable equals 1.

Section V: VARIABLE SPECIFICATION CARD(S)

Columns	
1- 4	1st variable
6- 9	2nd variable
76-79	16th variable

Variables are punched 16 to a card for as many cards as required. Both Computed variables (Cxxx) and Background variables (Vxxx) may be specified. Variables may be punched in any order, however, the program will reorder them low to high, Computed variables followed by Background variables. Background variables cannot exceed 6 characters (1A6) in size. A maximum of 100 variables combined (Cxxx and Vxxx) is allowed. C001, C002, C003 cannot be used.

PROGRAM IDENTIFICATION

Name	VARSUM (Variable Summary)
Language	FORTRAN V
Date Written	1966/May 1973
Programmer	CSC/Weissmuller

FUNCTION

This program computes and reports frequency distributions within specified intervals, makes total frequency counts, and calculates means and standard deviations on selected background and computed variables for any group of individuals whose job description has been generated by JOBGRP or JOBSPC.

INPUT/OUTPUT

Input	a. HISTORY/KPATH file
	b. JOB DESCRIPTION file
Output	a. Printed report
	b. Report optionally added to the REPORT file.

CONTROL

Section I FILENAMES CARD

Columns	
1- 6	VARSUM
8-18	Filename of the HISTORY/KPATH file
20-30	Filename of the JOB DESCRIPTION file
32-42	Filename of the REPORT file.

Section II IDENTIFICATION CARD

Columns	
1- 6	VARSUM
8-15	Study identification
17	Number of PAGE HEADING cards in Section III. 0-6 cards are allowed.

Section III PAGE HEADING CARDS

Columns	
1-80	Any information desired.

Section IV REPORT ID CARD

Columns	
1- 6	REPORT ID
8	Blank = Add the report to the REPORT file. 1 = Do not add the report to the REPORT file.
10-57	REPORT TITLE

Section V SPECIFICATION CARDS (Any number of cards allowed.)

Columns	
8-11	Variable number of the form Cxxx or Vxxx.
13-18	Six character lower limit for history variables, left justified. Cannot be all blank. Must be equal in value or logically lower than the upper limit.
20-25	Six character upper limit for history variables, left justified. If the upper limit is blank, only exact matches to the lower limit are to be counted. The lower and upper limits define an inclusive range.

27-32	Size of the frequency intervals to be generated for numeric history variables. Presence of this field tells the program to convert upper limit and lower limit to integer and generate equal sized intervals between the limits. History data is converted from BCD to integer and counted in its proper interval. Upper limit of each generated interval belongs to the next interval.
34	Blank = Do not print counts for this card. 1 = Print the frequency distribution.
35	Blank = Do not calculate mean and standard deviation for the variable on this card. 1 = Calculate and print mean and standard deviation for those members included in the range. Counts of 'OTHER' are excluded from these calculations.
36	Blank = Do not print a subtotal of those counted. 1 = Print a subtotal marked with an asterisk for each group after printing the frequency counts.
37	Blank = Do not print the count of those values that fall outside the range specified. 1 = Print the count of 'OTHER'.
40-49	Lower limit for a computed variable, in the FORTRAN form F10.4 or equivalent. Field may be zero or negative but must be algebraically less than or equal in value to the upper limit.

50-59	Upper limit for a computed variable range, FORTRAN form F10.4 or equivalent. The lower and upper limits define an inclusive range.
60-69	Interval size, indicating that the range defined by the low and high limits is to be subdivided into equal sized intervals. If this field is blank, a single inclusive interval is defined by the low to high limits. This field must be of the form F10.4 or equivalent. It must be positive.
70-79	End interval factor; this value in FORTRAN form F10.4 (or equivalent) describes how far over the final interval a datum can go and still be counted in the interval. If this field is blank, then .0001 is used. For example, a range 0 to 100 by 10 would generate intervals 1 but less than 10, 10.000 but less than 20, etc. Final interval is 90.000 but less than 100.0001.
Section VI	END SPEC CARD
Columns	
1- 8	END SPEC
Section VII	REQUEST CARDS
	There is no restriction on number of REQUEST cards. They are reported ten at a time. The final group may contain less than 10 REQUEST cards.
Section VIII	END CARD
Columns	
1- 3	END

Sections II through VIII may be repeated any number of times if multi-reports are desired.

GENERAL DESCRIPTION

VARSUM obtains the communication region and history relative table from the HISTORY/KPATH file. Variable titles from the dictionary are transferred to a scratch file. Then the ID card, PAGE HEADING cards, and report ID card are read and saved. Then up to 10 REQUEST cards are read and saved in core. The JOB DESCRIPTION file is searched sequentially for an NMASK array corresponding to each tape sequence number given on the ten REQUEST cards. The HISTORY/KPATH file is read forward and each record, if it enters one of the ten requested groups, it is given to the variable counting routine. This counting routine converts alpha to numeric if necessary and performs the data comparisons to the intervals or limits given on each SPEC card as generated in the spec table. The frequency report is then printed. The program then reads ten more REQUEST cards and repeats the process. If there are less than 10 requests remaining, the program continues on that lesser number. This continues until an END card is read. Then the program seeks another ID card. If one is present, the entire program recycles. If not, the program terminates.

PROGRAM NOTES

1. A new frequency distribution is started each time the variable number in Column 8-11 changes.
2. Each sequential SPEC card which has the same variable number in Column 8-11 continues the previous frequency distribution in the printout. Exact match counts may be intermixed with fixed range counts. Neither of these can be mixed with generated interval counts because the latter involves additional summations for computing mean and standard deviation.

3. A lower limit must exist on every SPEC card (Column 13-18 or 40-49). Columns 13-18 are the determining field for the program to decide how many characters of alphanumeric history data are to be used and therefore must be left adjusted with leading zeros.
4. Limits punched in Columns 13-18, 20-25, 27-32 indicate history data (BCD data on tape) and must have a V in Column 8.
5. Limits punched in Columns 40-49, 50-59, 60-69 indicate computed variables only (Floating point data on tape) and must have a C in Column 8.
6. A lower limit only (Columns 13-18 and 40-49) implies that the data from tape must compare exactly to the limit if it is to be counted.
7. Both a lower and an upper limit but no interval size indicates a fixed range which includes the limits as end points and the data from tape must be within the range to count.
8. If an interval size exists then the data must be numeric. The size field is the determining field for the program to decide to convert BCD data to numeric. The interval size must be greater than zero if it is used.
9. When generating intervals, such as 0 to 20 by 5, the upper limit of each interval is not included in the interval except the final one.

   For example, the intervals generated would be 0 - 4 (less than 5)
   5 - 9 (less than 10)
   10 - 14 (less than 15)
   15 - 20 End point included.

10. Means and standard deviations can be computed only on generated interval numeric data (cc 27-32 or cc 60-69 greater than zero).

11. The count of 'OTHER', which are those values which do not fall into any interval, does not enter into the mean or standard deviation calculation.

12. To force a fixed numeric range in a mean and standard deviation calculation, the size field (cc 27-32 or cc 60-69) must be greater than the interval.

    Example:

    low = 10, high = 30, size = 21

    will create only 1 interval, that being 10 to 30 and allow the values to enter the standard deviation calculation. This is because sums and sums of squares are calculated only on interval data which is indicated by a 'size' field > 0.

13. To force a single valued item to enter the mean and standard deviation calculation, the lower and upper limits are made equal and the interval size is punched > 0. Example: to count zero valued items and force them into the standard deviation calculation

    low = 00, high = 00, size = 01

    This is often used where zero data is a significant part of the mean, such as hours worked per week on a given task.

14. The lower and upper limits and size must be identical in length with the maximum number of BCD characters in a history datum to be counted. Example: a SPEC card with limits

    low = 1, high = 360, size = 10

    would count incorrectly since the low field indicates that only one character is to be inspected. In this example, the card should be punched.

    low = 001, high = 360, size = 010

15. For history data entering a generated interval calculation, the history data is considered unsigned numeric. No attempt should be made to calculate means or standard deviations on pure alphabetic data as it cannot be converted to meaningful numeric.
16. Computed variables 1 through 3 cannot be counted. They are control words.
17. Any number of REQUEST cards may be in the control deck. The program reads them ten at a time, and prints a report on all the variables for each ten REQUEST cards. This permits logical grouping of similar groups for visual comparison.
18. To change PAGE HEADING cards between REQUEST cards, control sections II, III, IV, VI, VII and VIII must be repeated, but once the SPEC cards Section V are entered into memory from the first deck, they remain there until new SPEC cards are encountered. If new SPEC cards are read, then all the previous set of SPEC cards is erased.
19. REQUEST cards are processed in input order and printed left to right across the page. Page titles are printed on the beginning page of every 10 REQUEST cards.
20. Each complete report, beginning with a VARSUM card and terminating with an END card, is considered one report on the output file and begins page numbering at one.
21. A new REQUEST card is punched for each END card encountered.
22. Size Restriction on SPEC DATA. The maximum number of variable specifications that can be computed at one pass is a function of the total number of generated intervals.

    Let I = total number of all intervals on all SPEC cards, including exact matches, discrete intervals and generated numeric intervals.

N = number of generated numeric intervals; example 0 to 100 by 10 generates 10 intervals.

R = number of REQUEST cards (10)

V = number of different variables.

then $5*I+R(I+2*N)+15*V \le 12500$

Example: If the SPEC cards on 50 different variables define 100 matches and in addition, generate 250 numeric intervals for matching, and there are 6 REQUEST cards, then

I = 100 + 250 = 350

N = 250

R = 6

V = 50

$5*350 + 6*(350+2*250) + 15*50 = 7600 < 12500$ and deck will run.

For extremely large variable SPEC card decks, it may be wiser to reduce the number of REQUEST cards per report rather than reduce the SPEC card deck.

MESSAGES

1. 'INVALID VARSUM CONTROL CARD.' Deck out of order or incorrect. Job aborted.
2. 'FOLLOWING CARDS NOT PROCESSED DUE TO PREVIOUS ERROR.' Cards listed are skipped and do not enter in the reports. They must be rerun.
3. 'TOO MANY PAGE HEADING CARDS.' Max of 6 cards permitted.
4. 'SID OR MID DOES NOT MATCH TAPE, JOB KILLED.' Mount correct HISTORY/KPATH file so that the identification matches that punched on the ID card.

5. 'NO REPORT IDENTIFICATION.' Columns 1-6 of report ID card are blank. Job must be rerun.

6. 'INVALID FIELD IN THIS SPEC CARD.' Correct the SPEC card and rerun. Error conditions most likely to occur are:

   (1) Bad number in column 9-11.

   (2) Column 9-11 greater than number variables on history tape.

   (3) If column 8 is a C, 9-11 must be greater than 003.

   (4) Lower limit blank.

   (5) Lower limit algebraically greater than upper limit.

   (6) Negative interval size.

   (7) Generated interval limits low or high contain alphabetic.

   (8) Size interval field zero or contains alphabetic.

   (9) Column 34-37 are all zero.

7. 'NO SPEC OR REQUEST CARDS.' Self-explanatory. Deck probably out of order.

8. 'BAD SID OR TSEQ OR RPTID. THIS CARD IGNORED.' Fields on REQUEST card do not match the JOB DESCRIPTION file. Card omitted from balance of report.

9. 'CANNOT FIND ALL TAPE SEQ NUMBERS FOR THIS VARSUM SET. THIS SET IS SKIPPED.' Probably bad JSEQ in REQUEST card. Cannot match the JOB DESCRIPTION file. May be bad punching or wrong file.

10. 'MEMORY CAPACITY EXCEEDED. REDUCE NUMBER OF SPEC CARDS. XXXXXX CELLS FOR VARIABLES ARE NEEDED IN THIS RUN PLUS YYYYYY CELLS TOTAL FOR TITLES.' SPEC cards, generated intervals or REQUEST cards caused overflow of current tables. Max size is 12,500 cells total including room for variable titles. Reduce number of SPEC cards or REQUEST cards on next run.

11. 'CANNOT CALCULATE MEAN OR STD DEV EXCEPT ON NUMERIC INTERVALS.' For means and standard deviation, the size interval field must be non-zero to indicate numeric conversion of history data. Means or standard deviation are meaningful only for such intervals. On pure alphabetic data, the size field must be blank.

12. 'MATRIX ID MISMATCH XXXXXX NOT = XXXXXX OF XXXXXX.' The REQUEST card and associated job description was not produced from the file which is currently mounted.

FORMAT CARDS

These cards describe the SETCHKed raw data file. There must be a format card to describe each raw data card within a case. If a case consists of 20 raw data cards, then 20 format cards are required. There are no restrictions on the layout of raw data cards or format cards except that each raw data card must contain a control number in columns 1-4 and each format card must contain 'C...' in columns 1-4. Raw data fields are defined by the following format characters:

C	Defines the beginning of the case control number. This character must appear in column 1 of each format card.
H	Defines the beginning of a history variable.
T	Defines the beginning of a task variable.
	Defines a continuation of any of the above. A task variable may occupy no more than 6 columns (T.....). The length of task variables (in terms of columns) must remain constant for all tasks.
b̸	A blank column defines a skip.
other	Any other character is defined as a 'check' character. A check character must have a matching character in the corresponding column of the raw data. A maximum of 500 'check' characters is allowed.

SAMPLE SELECTION

Section I VARIABLE INTERACTION CARD

Columns

2 Number of variable ranges used. (Maximum of 9) This also specifies the number of cards in Section II.

4-80 Variables and operators optionally nested in parens.

OPERATORS - Two logical operators may be used. These will perform a logical "AND" or a logical "OR". These operators are denoted by the following special characters:

Logical "AND" is denoted by "&".

Logical "OR" is denoted by ":" (Colon).

VARIABLES - The variable names may be abbreviated to least number of significant characters. For example, C007 may be written "C7", V020 may be written "V20", etc.

PARENS - If used, the parens must appear in matched pairs. The use of parens will cause the enclosed expression to be evaluated prior to all lesser or non-enclosed expressions. The normal method of evaluating the logical expressions is to evaluate all expressions containing "&" (which is the logical "AND") from left to right, then evaluate all expressions containing ":" (which is the logical "OR"). In other words, a logical "AND" has precedence over a logical "OR", as is the case in most MATH or LOGIC

Systems. Parens may be used for readability or to override the normal hierarchy.

Section II VARIABLE SPECIFICATION CARD (one card per variable)

Columns	
2	1 = The variable is 1-6 characters in size 2 = The variable is 7-12 characters in size blank = The variable is computed (Cxxx)
4- 7	Variable identification (Vxxx or Cxxx)
9-80	Variable low and high ranges. Values must be left adjusted in their fields. Both low and high must be specified. If cc 2 is 1: Punch low in cc 9-14, high in cc 15-20. If cc 2 is 2: Punch low in cc 9-20, high in cc 21-32. If cc 2 is blank: Same as 2 except values must contain a decimal point.

EXAMPLES

If C007 = Number of non-zero task responses, V003 = Grade.

```
3  C7&V003:V3
   C007  400.  683.
1  V003  1  3
1  V003  4  9
```

Then this setup will select all Airman First and below who perform between 400 and 683 tasks and all Sergeants and above regardless of the number of tasks they perform.

END

DEPT. OF HEW

NAT'L INSTITUTE OF EDUCATION

ERIC

DATE FILMED

AUG 28 75